МОЙ КОМПЬЮТЕР

#39 314

27.09-04.10.2004

сентябрь

Софт-пробирка # Недреманное око. Юзер спит, а web-мониторинг идет. **стр. 28**

Программирование # С машиной на «ты». Ассемблер – язык асов. **стр. 40**

Живая теория # Биокомпьютер – реальность и фантастика. Не секрет, что компы не растут в огороде... **стр. 18**

стр. 14

Intel Developer FORUM

В принципе важно
Экземпляры всех номеров газеты хранятся в лучших библиотеках Франции, Англии, Германии, США и в частных коллекциях.
На территории нашей страны издания «Мой компьютер» можно подписаться в ближайшем почтовом отделении, индекс 35327

ВЫХОДНЫЕ ДАННЫЕ

Всеукраинский еженедельник
«МОЙ КОМПЬЮТЕР» №39,
27.09.2004. Тираж: 18 500.
Рег. свидетельство: серия КВ № 3503 от 01.10.98.
Подписной индекс в каталоге «Укрпочта»: 35327.
Учредитель: ООО «К-Инфо».
Издатель: Издательский дом «Мой компьютер»
Киев, ул. Качалова, 6
info@mycomputer.ua
www.mycomputer.ua

Редакция: Киев, ул. Качалова, 6, тел. (044) 455-3575
Для писем: 03126, Киев-126, а/я 570/8
Издатель: Михаил Литвинюк.
Главный редактор: Татьяна Кохановская.
Зам. главного редактора: Сергей Мишко.
Железный редактор: Владимир Сирота.
Редакторы: Олег Касич, Игорь Ким.
Художественный редактор: Андрей Шмаркатюк.
Музыкальный редактор: Виктор Пушкар.
Эпистолярный редактор: Трурль.
Литературные редакторы:
Анна Китаева, Данил Перцов.
Верстка: Сергей Овсяник.
Художники: Федор Сергеев, Елена Маслова.
Корректор: Елена Харитоненко.
Разработка дизайна: © студия «J.K.™Design»,
Николай Литвиненко.
Отдел маркетинга: Надежда Николаева,
Роман Бураковский.
Реклама: Олег Федоров,
Валентина Маркевич-Кравченко.
Офис-менеджер: Тамара Задворнова.
Сбыт: Лариса Остаповская,
Елена Назарова, Михаил Ковальчук.
Начальник отдела полиграфии: Дмитрий Мажаев.
Экспедирование: Анатолий Клочко.
Разработка Web-сайта:
© Николай Угаров. (xKO).
Поддержка Web-сайта: Ростислав Стрелковский.
Пред. Издательского дома в Харькове:
Вячеслав Белов (viacheslavb@ua.fm)
Техническая поддержка: ISP «IT-Park»
Фотовывод: ООО «Мира» тел: (044) 247-4438
Печать: Типография ТМ «Мандарин»,
ТзОВ «Видавнича група "Експрес"» (Львівська обл.
Яворівський р–н, с. Рясне Руське, вул. Свободи 5
тел.: (0322) 97-4768)
Зак № 2053
Печать обложки: Типография «День Печати»
тел.: (044) 559-2655
Цена договорная.

ВНИМАНИЕ, ПРОМОКАЦИЯ

ак

Условия конкурса на странице 4

ОГЛАВЛЕНИЕ

01 Сергей ПАРИЖСКИЙ
Социальный взлом
Социальная инженерия и методы противостояния ей.
стр. 12–13 1

02 Сергей Н. МИШКО
Серенада Силиконовой Долины
Репортаж с IDF.
стр. 14–16 2

03 Александр КОНДАУРОВ
На витрине: AOpen i915Pa-E
Системная плата на чипсете 915P.
стр. 17 3

04 Евгений ФИДЕЛИН
Биокомпьютер — реальность и фантастика
Каковы перспективы органических компьютеров?
стр. 18–20 4

05 Алексей САЛО
Что у нас в карманцах, моя пр-релесть?
FAQ по КПК.
стр. 21–22 5

06 Виталий ЯКУСЕВИЧ
BIOS и его настройки
PCI TRDY Timer и пр.
стр. 23, 26 6

07 Олег КАСИЧ
Удачное знакомство
Продукты Fujitsu Siemens, Fujitsu и AOC.
стр. 24 7

08 Эдуард ОЧЕРЕТНЫЙ
Пингвин под окном
Дистрибутив Linux, запускающийся из-под Windows.
стр. 26–27 8

09 Марина и Сергей БОНДАРЕНКО
Недреманное око
Website-Watcher, утилита для мониторинга сайтов.
стр. 28–29 9

10 Надежда БАЛОВСЯК
Цифры в образах
Пакеты визуализации данных.
стр. 30–31 10

11 Павел ДМИТРИЕВ
ТеХничная верстка
ТеХника красной строки.
стр. 32–33 11

12 Дмитрий ПАВЛОВ
ИNSталляцию строIм Sами
Создаем дистрибутивы по своему вкусу.
стр. 34–37 12

13 Владислав ДЕМЬЯНИШИН
Мысли о Паскале
ООП в Паскале.
стр. 38–39, 43 13

14 Nick 'nVr'
С машиной на «ты»
Начинаем знакомство с основами Ассемблера.
стр. 40–41 14

15 Ольга КАЛИТКА
Больные и их болезни
Несерьезный медицинский справочник.
стр. 42–43 15

16 Трурль
Беседка «Моего компьютера»
Конкурсы настоящие и будущие.
стр. 44–45 16

Для участия в конкурсе впишите свои данные:

Ф. И. О. ____________

Почтовый адрес ____________

Телефон ____________

ВНИМАНИЕ!

Места, где Вы всегда можете приобрести издания ИД «Мой компьютер» — журнал **«Реальность фантастики»**, а также еженедельники **«Мой компьютер»** и **«Мой компьютер игровой»**:

Винница
- ✓ Магазин «Світ книги», ул. Келецкая
- ✓ Лоток на углу Коцюбинского и Ленинградской

Днепропетровск
- ✓ Киоски «СВ-почта»

Донецк
- ✓ Киоски «Союзпечать»
- ✓ Магазин «Мир прессы», ул. Горького, 59-а, тел. 3853960
- ✓ ул. Артема, 131-а
- ✓ ул. Освобождения Донбасса, 4

Макеевка
- ✓ гост. «Маяк»

Киев
- ✓ Киоски «Союзпечать»
- ✓ Торговые точки «CN-Столичные новости»
- ✓ Киоски «Факты»
- ✓ Книжный рынок «Петровка»
- ✓ Книжный супермаркет «Буква»
- ✓ Сеть книжных магазинов и торговых точек «Орфей»
- ✓ Книжный магазин «Сучасник», пр. Победы, 29
- ✓ ст. м. «Лесная», остановочный комплекс
- ✓ ул. Жилянская, 87/30

Крым
- ✓ **Севастополь** — киоски «Союзпечать»

Луганск
- ✓ Магазины и киоски «Луганскпечать»

Львов
- ✓ Киоски «Торгпресса»
- ✓ Киоски «Интерпресса»

Мариуполь
- ✓ Киоски «Союзпечать»

Николаев

Торговые лотки:
- ✓ ул. Советская
- ✓ Супермаркет «Сельпо»
- ✓ ул. Комсомольская, возле клуба «Мужество»
- ✓ рынок на ул. Дзержинского
- ✓ рынок «Северный»
- ✓ «Саммит-Николаев», ул. Космонавтов, 61, тел. 581217

Одесса
- ✓ киоски «Одессагорпресса»
- ✓ киоски «Пресс-служба Одессы»

Оптовая продажа:
- ✓ ул. Костанди, 100

Полтава
- ✓ киоски Полтавского почтампта
- ✓ газетный ряд «Анюта», ул. Октябрьская, 27
- ✓ лоток на ост. «Оптика» (м-н «Осень»), ул. Ленина, 118

Сумы
- ✓ Укрпочта

Тернополь
- ✓ лотки «Газеты, журналы, кроссворды»

Харьков
- ✓ газетный рынок
- ✓ магазин «BOOKS»

Херсон
- ✓ киоск, бул. Мирный, 5
- ✓ киоск, ул. Железнодорожная

Хмельницкий
- ✓ Оптовая продажа (0382) 795668

Черновцы
- ✓ киоски «Укрпочта»

ПОДПИСКА – 2004

☞ Подписаться на **«Мой компьютер»** можно во всех отделениях «Укрпочты», индекс по каталогу 35327. Стоимость издания, в зависимости от периода, составляет: **1 месяц** – 10.34 грн, **2 месяца** – 20.80 грн, **3 месяца** – 30.72 грн, **4 месяца** – 40.88 грн, **5 месяцев** – 50.80 грн, **6 месяцев** – 60.72 грн. **7 месяцев** – 71.24 грн, **8 месяцев** – 81.16 грн, **9 месяцев** – 91.08 грн.

☞ Кроме того, работают следующие сайты с on-line предоплатой: **www.poshta.kiev.ua**, **www.blitz-poss.com.ua**, **www.kss.kiev.ua**, и для жителей зарубежья — **www.ukrpressa.kiev.ua.**

☞ Подписку с курьерской доставкой можно осуществить через следующие фирмы:

Киев
Саммит* 254-5050,
KSS* 464-0220,
Блиц-информ* 518-6682
(* филиалы по всем областным центрам Украины)
Периодика* 228-6165

Днепропетровск
Меркурий (056) 744-7287

Донецк
Идея (062) 381-0930,

Запорожье
Пресс-сервис (0612) 62-5151

Кременчуг
Саммит-Кременчуг (05366) 3-2188
Приватна доставка (05366) 2-5833

Львов
Деловая пресса (0322) 70-5482,
ЧП Циндра 97-1515,
Львовский курьер 21-2201
Саммит-Львов (0322) 74-3223

Николаев
Ноу-хау (0512) 47-2003
Саммит-Николаев (0512) 56-1069

Одесса
МиМ (0482) 37-5264

Севастополь
Истар (0692) 71-6219
(филиалы во всех городах Крыма)

Симферополь
Клуб бухгалтеров (0652) 27-2019
Саммит-Крым (0652) 51-2493

Харьков
Саммит-Харьков (0572) 14-2260

Херсон
Кобзарь (0552) 22-5218

Червоноград
Пресс-курьер (03249) 2-2250
От А до Я (03249) 2-9117

☞ Приобрести **«Мой компьютер»** в розницу можно в киосках и на раскладках по всей территории Украины.

УСЛОВИЯ КОНКУРСА

«АКТИВНО ВЕЗУЧИЙ ЧИТАТЕЛЬ»

1. В конкурсе участвуют все письма читателей, проставивших оценки по 10-балльной шкале всем статьям, указанным в оглавлении.
2. Нужно просто выслать вырезку из газеты с проставленными оценками статей в оглавлении номера (см. на обороте). Электронные письма в конкурсе не участвуют.
3. Если вы присылали письма к каждому номеру месяца (но не более 1 на номер), все они будут участвовать в розыгрыше призов среди читателей, то есть ваши шансы увеличиваются в 4 раза!
4. Вместе с подведением итогов конкурса «ЛУЧШАЯ СТАТЬЯ МЕСЯЦА» разыгрываются 1 первый, 2 вторых и 3 третьих приза среди читателей.



ИНТЕРНЕТ

Кое-что о круглых цифрах

Количество пользователей, зарегистрированных в **почтовой службе Mail.Ru** с момента ее открытия, приближается к 20 миллионам. Компания Mail.Ru собирается отметить это знаменательное событие. Двадцатимиллионный клиент почтовой службы Mail.Ru получит суперприз — автомобиль ВАЗ 2110. Точно такой же автомобиль получит и лучший друг победителя по переписке.



При регистрации почтового ящика новый пользователь должен в специальном поле указать имя, фамилию и адрес на Mail.Ru своего друга, если он хочет, чтобы тот тоже участвовал в акции. Тогда в случае, если он станет счастливым двадцатимиллионным пользователем Mail.Ru, друг тоже получит в свое распоряжение точно такую же машину.

По прогнозам аналитического отдела компании, двадцатимиллионный аккаунт будет зарегистрирован в период с 9 сентября по 27 октября, в зависимости от активности притока новых пользователей. Но имя счастливчика будет объявлено только по окончании этого периода, а именно 27 октября в 13:00 по московскому времени.

Кроме того, целый месяц, с 27 сентября по 27 октября, среди участников акции будут проводиться ежедневные розыгрыши призов, ценность которых будет расти с каждым днем. Более подробную информацию о конкурсе можно найти на страничке **http://20mln.Mail.Ru**.

Источник: *Компьюлента*

Поиск на любой вкус

Поисковая система **Ask Jeeves**, не отставая от последних новшеств ведущих поисковиков Google и Yahoo, представила свой собственный продукт **MyJeeves**. Этот новый сервис позволит пользователям вести историю своих поисков, персонализировать запросы. MyJeeves предоставляет возможность сохранять найденные веб-страницы, а потом разбивать сохраненные документы по папкам, созданным по усмотрению пользователя. Количество сохраняемых документов для зарегистрированных пользователей не ограничено, для незарегистрированных — лимитировано тысячей документов на одном компьютере. Таким образом, использовать найденную информацию станет удобнее из-за возможности ее организации и сортировки.

Источник: *Компьюлента*

Ослик приостал...

Рыночная доля альтернативных браузеров выросла на 6.3%. Согласно последним опросам, проведенным аналитиками WebSideStory, доля браузера компании Microsoft понизилась на 1.8% за три месяца. Теперь Internet Explorer занимает 93.7% рынка браузеров. Такое падение в период с июня по июль обусловлено, в первую очередь, серьезными проблемами с безопасностью. Опрос обозначил две категории альтернативных браузеров: к первой относятся *Netscape* и *Mozilla*, ко второй — все прочие браузеры, включая Opera. О росте популярности обеих категорий говорят цифры: первая категория получила 5.2% рынка, вторая — 1.1%.

Источник: *Компьюлента*

Европа vs кибермафия

На трехдневной конференции, организованной Советом Европы и состоявшейся на прошедшей неделе, делегации из США, Австралии, Китая и стран ЕС обсуждали и разрабатывали методы совместной борьбы с киберпреступностью, сообщает Associated Press. Основной темой конференции был вопрос об объединении усилий правительств разных стран, для того чтобы остановить рост числа онлайновых преступлений. На сегодняшний день основные виды киберпреступлений — мошенничество и нарушение авторских прав.

Принятая Советом Европы *Конвенция о киберпреступности*, являющаяся первым международным договором такого рода, в 2001 году была подписана 30 странами, включая Канаду, Японию, ЮАР и США. Албания, Хорватия, Венгрия, Румыния, Словения, Македония, Эстония и Литва являются единственными государствами, ратифицировавшими договор, в котором указываются четыре типа киберпреступлений: взлом компьютерных систем, мошенничество, запрещенный контент (расистские сайты и веб-страницы с детской порнографией) и нарушение авторских прав.

В сводном отчете, представленном на конференции, было отмечено, что темпы роста числа онлайновых правонарушений увеличиваются с каждым годом.

Согласно обнародованным данным, в Германии на долю онлайновых преступлений приходится 1.3% от общего количества правонарушений, однако киберпреступностью было нанесено 57% всего материального ущерба, или 8.3 млрд. долларов США. Исследование, проведенное в США в 2004 году, показало, что 20% из 494 американских корпораций подверглись хакерским атакам или пострадали от фишинга. Количество веб-сайтов, разжигающих межнациональную и межрасовую рознь, пропагандирующих жестокость и насилие, выросло с 2000 года на 300%. А доходы от онлайновой продажи детской порнографии в текущем году составили 20 млрд. долларов США. Организованная преступность, прочно укоренившаяся в киберпространстве, использует Интернет для совершения экономических преступлений.

Источник: *Компьюлента*

ПРОГРАММЫ

Легальное ПО — в кредит

15 сентября компания **«Майкрософт Украина»** совместно с **ПроКредит Банк** и **Райффайзенбанк Украина** провели пресс-конференцию, на которой объявили о старте *совместной программы кредитования субъектов малого и среднего бизнеса на приобретение легального ПО.*



Необходимость создания такой программы была обусловлена целым рядом факторов. Прежде всего, компания «Майкрософт Украина» учитывала тот факт, что, как показало исследование GfK-USM, владельцы малого и среднего бизнеса, как правило, не располагают необходимым количеством денежных средств для покупки легального ПО, что вынуждает их использовать политику пастепенной легализации — когда на протяжении продолжительного времени предприятие использует как легальные копии, так и нелегальное ПО. При этом, используя нелегальное ПО, компании ставят под угрозу сохранность своих данных, внутреннюю безопасность, подвергают себя техническим и юридическим рискам.

«Сегодня существуют серьезные препятствия для быстрой легализации программного обеспечения в малом и среднем бизнесе», — прокомментировал актуальность данной программы директор компании «Майкрософт Украина» Валерий Лановенко. «Далеко не все предприятия могут единоразово инвестировать необходимые суммы в легальное ПО для обеспечения технической безопасности и юридической чистоты своего бизнеса. Запуском настоящей программы мы открываем уникальный для Европы проект, который, как мы надеемся, поможет украинскому малому и среднему бизнесу повысить эффективность своих инвестиций в информационные технологии».

Срок действия программы — с 15.09.2004 по 15.12.2004 г. Основным условием предоставления кредита является наличие предложения на поставку программного обеспечения от уполномоченного партнера Microsoft Corporation, а также заполненных и заверенных печатью предприятия результатов инвентаризации используемого программного обеспечения (может проводиться как уполномоченным партнером Microsoft Corporation, так и самостоятельно), плана легализации программного обеспечения и утвержденных правил использования программного обеспечения на предприятиях.

Указанные формы должны быть заполнены по установленным шаблонам, предоставляемым компанией «Майкрософт Украина» и уполномоченными партнерами Microsoft Corporation (корпорация *«Инком»*, карпорация *«Квазар-Микро»*, ЗАО *«Софтлайн»*, компания *Украинский Центр Лицензионного Программного Обеспечения*, компания *Е-консалтинг*). Образцы требуемых форм, а также контакты уполномоченных партнеров Microsoft Corporation можно найти на сайте **www.microsoft.com/ukraine/credit**.

Условия кредитования, предлагаемые по программе, были разработаны в соответствии с кредитной политикой ПроКредит Банка и Райффайзенбанк Украина.

Райффайзенбанк Украина предоставляет кредиты как юридическим лицам, так и частным предпринимателям (опыт предпринимательской деятельности должен быть не менее 24 месяцев). Сумма кредита — эквивалент от $5000 до $60 000, валюта кредита — доллары США, евро, гривны. Процентные ставки по кредиту — 13%, если кредит предоставляется в долларах США или евро, 18% — если в гривнах (такая ставка предлагается банком только для кредитов по данной программе, стандартная процентная ставка Райффайзенбанк Украина по кредитам в гривнах — 19%). Обязательным условием кредитования является предоставление недвижимости в качестве залога. Среди критериев принятия решения о предоставлении кредита на легализацию ПО Райффайзенбанк Украина выделяет платежеспособность заемщика, его финансовую стабильность и достаточность обеспечения кредита.

Условия, предлагаемые для программы ПроКредит Банком, предполагают кредитование без внесения залога для суммы кредита, не превышающей 30 000 гривен (кредит предоставляется без залога, но с одним Поручителем). Обычно беззалоговые кредиты (экспресс-кредиты) выдаются банком на сумму до 15 000 гривен. Для сумм кредита свыше 30 000 гривен предусмотрено обеспечение согласно требованиям кредитной политики банка. Максимальная сумма кредита, предоставляемого ПроКредит Банком — 300 000 гривен, кредит может предоставляться в гривнах либо долларах США с конвертацией в гривны. Размер процентной ставки определяется в соответствии с установленными тарифами банка для соответствующих сумм кредитов. Схема погашения кредита — ежемесячно равными частями.

Максимальный срок кредитования по программе — 12 месяцев со дня выдачи кредита. Для предоставления дополнительной информации и ответов на вопросы, связанные с работой программы, открыта «горячая линия»: 8-800-504-44-40 (круглосуточно, бесплатно по всей Украине).

Источник: *«Майкрософт Украина»*

Троянец-пачкун

Компания *Sophos* предупреждает о появлении новой вредоносной программы **Autex**. Как сообщается, троян Autex способен заражать компьютеры, работающие под управлением операционных систем Microsoft Windows. После активации Autex создает на жестком диске инфицированной машины свои многочисленные копии, а также «захламляет» винчестер файлами в формате HTML с произвольными именами. Кроме того, вредоносная программа прописывает себя в ключе автозапуска реестра и изменяет ряд настроек браузера Internet Explorer. Каких-либо деструктивных функций в коде Autex специалисты Sophos пока не обнаружили. Дополнительную информацию о трояне ищите на странице **http://www.sophos.com/virusinfo/analyses/trojautexa.html**.

Источник: *Компьюлента*

Вести с linux-фронтов

Компания *ALT Linux* выпустила обновленную версию одного из своих дистрибутивов — **ALT Linux 2.3 SOHO Server**. SOHO Server включает в себя стандартный набор серверных приложений: apache, squid, postfix, samba и bind. По словам разработчиков, упор в версии 2.3 был сделан на улучшение функциональности дистрибутива в роли межсетевого экрана. SOHO Server вполне пригоден для использования в роли брандмауэра для частной или корпоративной сети, защищающего внутреннюю сеть от проникновений извне. В пресс-релизе ALT Linux подчеркивается, что SOHO Server обладает развитыми средствами администрирования (в том числе и удаленного) и прост в настройке.

Источник: *Компьюлента*

Новый контрразведчик

Компания *Lavasoft* выпустила обновленную версию популярной программы **Ad-Aware SE Build**, предназначенной для поиска и правильного удаления spyware — следящего (шпионского) ПО, такого как *Parasites*, *Scumware*, *Trojans*, *Dialers*, *Malware*, *Browser hijackers* и т.д., устанавливаемого на ПК пользователя различными Ad-Aware программами.

Ad-Aware сканирует память, реестр и жесткий диск ПК на предмет обнаружения «паразитов» и затем удаляет их. Имеется возможность бэкапа и отката (для случаев, когда установившая шпиона программа отказывается жить без оного), многоязыковая поддержка, возможность интеграции в Windows-shell, игнор-лист и возможность работы из командной строки. Для облегчения пополнения базы spyware существует бесплатный Add-on Ref Update, который находит свежие Reference File, загружает и устанавливает их.

В текущем выпуске добавлены новые параметры командной строки, добавлена поддержка UNC, улучшено логирование информации, добавлены новые опции и т.д.

Скачать Ad-Aware SE Build v.1.05 можно по ссылке **http://files.savagenews.com/aawsepersonal-1.05-sn.exe** (2,5 Мб, Shareware, Windows All).

Источник: *IXBT*

ТЕХНОЛОГИИ

Краеугольный камень

Американская компания **AMD** официально представила процессор **Mobile Athlon 64 3000+**, предназначенный для использования в портативных компьютерах. Чипы изготавливаются по 90-нанометровой технологии и ориентированы прежде всего на сектор мощных ноутбуков.

Системы на базе Mobile Athlon 64 3000+ смогут работать как со стандартными 32-битными приложениями, так и с 64-разрядными программными пакетами. Новые процессоры поддерживают шину HyperTransport и технологию энергосбережения AMD PowerNow!, продлевающую время автономной работы портативных устройств от аккумуляторов. Кроме того, в кристаллах реализована фирменная антивирусная защита *Enhanced Virus Protection (EVP)*. Правда, технология EVP в настоящее время поддерживается только лишь операционной системой Microsoft Windows XP со вторым пакетом обновлений.

Первым ноутбуком, построенным на базе процессора Mobile Athlon 64 3000+, стал **Ferrari F3400** от компании **Acer**. Лэптоп оборудован жидкокристаллическим дисплеем с диагональю 15" (разрешение 1400×1050 пикселей), 512 Мб оперативной памяти, жестким диском емкостью 80 Гб, DVD-рекордером и мощным графическим контроллером ATI Mobility Radeon 9700 со 128 Мб видеопамяти. Стоимость портативного компьютера установлена на уровне $2400. Что касается самих чипов Mobile Athlon 64 3000+, то их поставки уже начались. Цена процессоров в партиях от одной тысячи штук составляет $241.

Источник: *Компьюлента*

Централизация графики

Компания **ATI Technologies** представила **HyperMemory** — технологию, позволяющую снизить стоимость ПК за счет использования графическими процессорами системной памяти для собственных «графических» нужд. HyperMemor использует возможности высокоскоростной двунаправленной передачи данных *PCI Express* для хранения и доступа к графическим данным, находящимся в системной памяти, таким образом снижая зависимость систем от объема графической памяти (впрочем, это остается сомнительным в виду перехода индустрии на DDR2, высоких цен данной памяти на начальном этапе и необходимости увеличения объема системной памяти еще минимум на 256 Мб).

Как отмечается в пресс-релизе, ранние стандарты обеспечивали относительна низкую скорость передачи данных между графическим процессором и центральным процессором — для графических приложений, работавших в масштабе реального времени. Поэтому видеокарты поставлялись с 256 Мб собственной памяти для хранения текстур и данных для рендеринга сцен. Технология HyperMemory позволяет ATI и ее партнерам поставлять видеокарты с уменьшенным объемом видеопамяти.

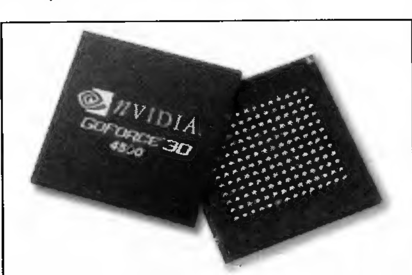
Технология использует алгоритмы «интеллектуального выделения памяти» для оптимизированного использования дос-

тупной локальной памяти (памяти видеокарты). «Оптимальность» размещения данных в системной или «локальной» памяти определяется динамически. Отмечается, что видеокарты, использующие технологию HyperMemory, будут представлены в этом году.

Источник: *iXBT*

Карманная мощь

На выставке *Mobile Content World*, проходящей в Лондоне, компания **NVIDIA** представила новый графический процессор **GoForce 3D 4500** для карманных беспроводных устройств.



Новинка, рассчитанная на установку в мобильные телефоны и карманные компьютеры, является первым серийным ускорителем трехмерной графики для мобильных устройств с программируемыми пиксельными шейдерами. Чип способен одновременно накладывать на каждый пиксель до шести текстур, обеспечивает восьмикратное цифровое увеличение, имеет аппаратные кодеки MPEG4 с возможностью одновременного кодирования и декодирования видео, а также JPEG-кодек. 64-разрядное ядро для обработки двухмерной графики ускоряет вывод на экран цветных изображений.

Процессор поддерживает работу с флэш-картами стандартов Secure Digital и MMC, включая карты расширения SDIO, в том числе модули фотокамеры и беспроводной связи стандартов Bluetooth и Wi-Fi. Чип имеет 1280 Кб встроенной памяти (предыдущие модели семейства GoForce имели от 160 до 640 Кб памяти) со 128-разрядным интерфейсом, что дает возможность без использования прочих ресурсов мобильного устройства поддерживать дисплеи с разрешением до 640×480 пикселей.

Новая микросхема обеспечивает высокое качество изображения в компьютерных играх, а также при просмотре цифровых фотографий и видеороликов. Как утверждает разработчик, по реалистичности изображения мобильные устройства с процессором GoForce 3D 4500 могут сравниться с настольными ПК.

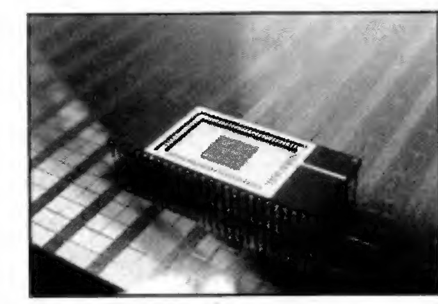
Встроенный геометрический процессор позволяет снизить нагрузку на центральный процессор мобильного устройства, а реализованная в GoForce 3D 4500 технология энергосбережения NVIDIA nPower способна продлить время автономной работы портативной системы.

Процессор производится в корпусах BGA с 168 выводами по КМОП-технологии с нормой 0.15 микрона; напряжение питания ядра — от 1.425 до 1.575 В; размеры микросхемы — 12×12×1.2 мм.

Источник: *Компьюлента*

Узелки на память

Компания **Samsung Electronics** заявила о ряде «прорывов в ключевых технологиях». Первое заявленное достижение — разработка первого в мире 60-нм 8-Гбит чипа NAND флэш-памяти. Массовое производство этих чипов начнется в конце следующего года, а в первом квартале 2005 г. намечено начало выпуска 4-Гбит чипов NAND флэш-памяти.



Бизнес Samsung в секторе NAND флэш-памяти вырос с $400 млн. в 2001 г. до $2.1 млрд. в прошлом году. В этом году компания планирует удвоить объемы продаж по сравнению с прошлогодними и занять 65% этого рынка в мировом масштабе.

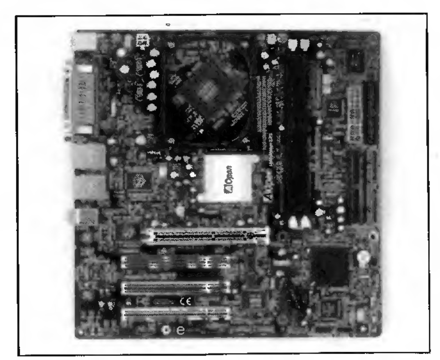
Вторым достижением объявлена разработка первого в мире 2-Гбит DDR2 SDRAM модуля с использованием существующего 80-нм техпроцесса. Компании удалось добиться такого результата путем совершенствования дизайна модуля с применением трехмерных транзисторов и собственно техпроцесса, вопреки бытующему мнению, что для изготовления модулей такого объема необходим переход на 65-нм или более тонкую технологию.

Samsung планирует начать массовое производство этих модулей во второй половине 2005 г. По прогнозу аналитической компании *Gartner Dataquest*, доля DDR2 выраствет с 11% в этом году до 50% к концу следующего, и она станет доминирующей на рынке DRAM.

Источник: *3DNews*

Недосол — на столе

Идея создания настольного компьютера на основе мобильного процессора уже далеко не нова. Между тем, до сих пор основным препятствием на пути ее реализации является стоимость, причем не процессоров, цена которых уже опустилась до приемлемого уровня, а материнских плат, обладающих поддержкой соответствующих мобильных чипов. Искренне хотелось бы надеяться на то, что со временем ситуация все-таки изменится.



Лишним подтверждением того, что это пожелание не является пустым сотрясанием воздуха, можно с уверенностью назвать появление на японском рынке Акихабара материнской платы **i855GMEm-LFS** производства **AOpen**. Ознакомимся с основными характеристиками новинки:

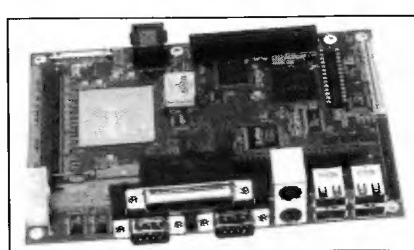
✓ поддержка процессоров Socket 479 Pentium M (Dothon/Banias) с FSB 400 МГц;
✓ чипсет: i855GME/ICH4-M;
✓ два слота памяти DDR333 (до 2 Гб);
✓ интегрированное графическое решение;
✓ один AGP-4x и три PCI-слота;
✓ по два ATA100- и Serial-ATA разъема (контроллер Promise PDC20579);
✓ два порта Gigabit Ethernet (Marvell 88E8001);
✓ шесть USB-2.0 и два FireWire-порта;
✓ интегрированное аудиорешение класса AC'97;
✓ форм-фактор: microATX.

Представленный набор возможностей, как нетрудно заметить, выглядит практически в полном соответствии с требованиями сегодняшнего дня (по крайней мере как для мобильного чипсета). Здесь и Serial-ATA, и Gigabit-Ethernet порты. Причем, наличие последних свидетельствует о том, что плата предназначена в том числе и для создания компактных серверов. К недостаткам следует отнести разве что наличие AGP-4x слота. Впрочем, вряд ли кому придет в голову делать из платформы на основе мобильного процессора «навернутую» игровую или графическую станцию. Перейдем теперь к самому интересному — стоимости материнской платы. Прогнозируемая цена платы в момент начала ее продаж (пока речь идет лишь о демонстрации новинки, в рознице она станет доступна в октябре) составит $270. Для сравнения: отдельные Socket-479 платы, появлявшиеся на рынке до сего времени, стоили не меньше $300–350. Согласитесь, разница очевидна...

Источник: *Ф-Центр*

На маленьком плоту

Компания **VersaLogic**, производитель одноплатных компьютеров форм-фактора 4.5"×6.5" серии *EPIC*, анонсировала новую модель компьютера **EPIC Gecko** для систем, используемых в медицине, промышленности, устройствах защиты информации и др. Одноплатный Gecko является полноценным компьютером и совместим с операционными системами Linux, Windows XP, Windows XP Embedded и другими встраиваемыми операционными системами.



Gecko построен на базе процессора x86 с низким энергопотреблением *AMD Geode GX-500*, процессор потребляет порядка 5 Вт и не требует активного охлаждения.

Основные параметры Gecko:

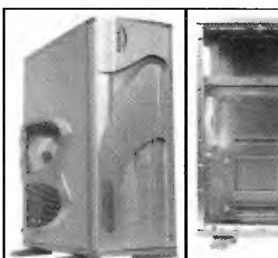
✓ процессор AMD GX-500 366 МГц;
✓ память — один слот для модулей памяти SODIMM (до 512 Мб DDR PC2100/PC2700);
✓ слот для карт расширения CompactFlash Type I/II;
✓ встроенное видео: AGP с поддержкой стандартных дисплеев CRT/LCD или индустриальных TFT-панелей;
✓ встроенный порт Ethernet 10BaseT/100BaseTX;
✓ управляемый загрузочный модуль Argon, с поддержкой протоколов PXE, RPL, NetWare, TCP/IP (DHCP, BOOTP);
✓ встроенный модуль АЦП, 8 каналов, 12 бит;
✓ встроенное аудио — кодек AC'97;
✓ четыре порта USB 1.1;
✓ IDE-интерфейс ATA-5, UDMA66;
✓ последовательные порты COM1/COM2 — RS-232, COM3/COM4 — RS-422/485;
✓ один параллельный порт;
✓ поддерживаются PS/2-клавиатура и мышь.

Ожидается, что этот одноплатный ПК поступит в продажу в четвертом квартале 2004 года по цене $650.

Источник: *3DNews*

Касательно винчестеров...

Компания **Maxtor** представила новое поколение внешних жестких дисков — **OneTouch II**, основной сферой применения которых является резервное копирование (собственно, функциональность устройства описана в его названии, поскольку сам процесс создания копии информации на жестком диске происходит по нажатии одной кнопки).



Как и во многих аналогичных разработках, в устройствах реализована функция парольного доступа к содержащейся информации; «закрытие» данных паролем происходит в момент отключения питания диска или системы.

Среди прочих возможностей OneTouch II в пресс-релизе компании упомянута функция подгрузки загрузки с диска Mac (Mac bootability function) — владельцы данных накопителей имеют возможность создавать «загружаемую» резервную копию системы, а также обновленный внешний вид НЖМД, «подходящий как для дома, так и для офисного использования устройства».

Среди кратких технических характеристик дисков упомянуты:

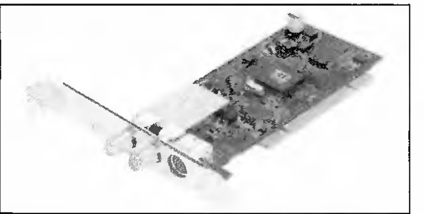
✓ скорость вращения — 7200 об/мин (250-Гб модель), 5400 об/мин (300-Гб модель);
✓ емкость — 250/300 Гб;
✓ буфер — 16 Мб;
✓ интерфейс — FireWire/USB 2.0;
✓ среднее время позиционирования — 9.0 мс;
✓ габариты — 41×140×210 мм;
✓ масса — 1.38 кг.

Рекомендованная цена — около $380 за 300-Гб модель, и около $330 за 250-Гб накопитель. В этом же году Maxtor планирует представить устройства других емкостей.

Источник: *iXBT*

Театр под Tulьм

Компания **Tul** сообщила о выпуске **PowerColor THEATER 550 PRO**, «домашнего театра», выполненного, как ясно из названия, на базе *ATI THEATER 550 PRO*. От обычных, представлявшихся ранее, ТВ-тюнеров компании новинка отличается наличием реализованных дополнительных технологий, включая 3D Comb Filter и Naise Reduction, позволяющих несколько повысить качество воспроизводимого изображения.



Стараясь охватить как можно более широкий спектр пользователей, Tul выпустила низкопрофильную карту, что позволяет использовать решение в системах в slim-корпусах. Мировой дебют PowerColor THEATER 550 PRO состоится в 4 квартале 2004 года; карта будет иметь рекомендованную розничную цену около $100.

Источник: *iXBT*

Вткидной экран

Японская корпорация **Sony** сообщила о выпуске нового карманного компьютера **Clie PEG-VZ90**, который поступит в продажу в конце текущего месяца. «Сердцем» устройства является ARM-процессор Sony с тактовой частотой 123 МГц. Наладонник оборудован 64 Мб оперативной памяти и 128 Мб флэш-памяти. Дисплей новинки, построенный на базе органических светодиодов, имеет диагональ 3.8", разрешение 480×320 пикселей и обладает очень высоким показателем контрастности — 1000:1 (угол обзора составляет 180°).



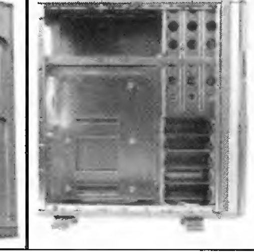
Особого внимания заслуживает расположение органов управления, которые спрятаны под сдвигающимся вверх экраном. Такая конструкция позволила уменьшить размеры карманного компьютера, не оказав при этом негативного влияния на удобство работы с устройством. Новинка также снабжена слотами Memory Stick Pro и Compact Flash, стереофоническими громкоговорителями и контроллерам Wi-Fi, обеспечивающим возможность подключения к беспроводным локальным сетям стандарта IEEE 802.11b. Предусмотрены также порт USB, инфракрасный порт и гнезда для подключения наушников.

Работает карманный компьютер Sony Clie PEG-VZ90 под управлением операционной системы Palm OS 5.2.1. Среди предустановленного программного обеспечения можно выделить медиаплейер, интернет-браузер и клиент электронной почты. Питается наладонник от ионно-литиевого аккумулятора емкостью 1300 мАч. Заряда батареи, согласно заявлениям производителя, должно хватить на четыре часа работы в режиме воспроизведения видео или на двенадцать часов непрерывного проигрывания MP3-композиций. Размеры устройства — 109×87×23 мм, вес — около 270 граммов. Стоимость модели Clie PEG-VZ90 составит порядка $860.

Источник: *Компьюлента*

Ловчая акула

Ассортимент компании **Thermaltake** пополнился еще одним корпусом. В полном соответствии с богатыми традициями *Xaser* и *Tsunami*, главным достоинством линейки под названием **Shark** (акула), несомненно, является крайне эффектный внешний вид, дополняемый синей неоновой подсветкой.



Впрочем, в части сборки корпус также не подкачал: впервые в продукции Thermaltake мы видим полностью съемную платформу для материнской платы, да и открывающаяся как влево, так и вправо передняя дверца тоже придает особую свежесть изделию. В остальном Shark вполне обычен (насколько это слово вообще может быть уместно в отношении продукции Thermaltake):

✓ размеры: 540×205×500 мм;
✓ материал: алюминий;
✓ отсеки: 7×3.5", 5×5.25";
✓ охлаждение: 2×120-мм вентилятора (2000 об/м, шум — 21 дБ);
✓ порты на передней панели: 2× USB 2.0, IEEE1394, микрофон/наушники;
✓ вес: 6.8 кг.

По словам представителей компании, черный и серебристый вариант исполнения попадут в продажу уже в следующем месяце, цена пока неизвестна.

Источник: *3DNews*

Адреса источников:
3DNews: http://www.3dnews.ru
Ф-Центр: http://www.fcenter.ru
Компьюлента: http://www.compulenta.ru
iXBT: http://www.ixbt.com

РЕДАКЦИОННЫЕ НОВОСТИ

Осенний посев

15 сентября нынешнего года студенты и преподаватели *Киевского национального университета им. Т. Г. Шевченко* получили прекрасный подарок, приуроченный к 170-летнему юбилею своей alma mater. Именно в этот день, в 10 часов утра, в торжественной обстановке была открыта **Лаборатория Microsoft**, получившая постоянную прописку в одной из аудиторий кибернетического факультета университета.

На открытии, в присутствии студентов, преподавателей и журналистов, выступил директор по стратегии Microsoft в Украине и России *Игорь Агамирзян*. Его доклад был посвящен новой волне развития информационных технологий. Докладчик не только рассказал о достижениях современности, но и сделал экскурс в историю, поведав собравшимся о том, как информационные технологии внедрялись в общество. Не обошел он и вопрос, который интересует многих: взаимозависимость операционных систем и компьютерного «железа».

Благодаря появлению Лаборатории, будущие специалисты в области IT получили возможность пользоваться самым передовым ПО. Резонно предположить, что, помимо использования в повседневной практике новейших лицензионных ОС Windows и приложений к ним, кибернетики КНУ смогут испытать свои силы и в их усовершенствовании. Не исключено, что самые талантливые «экспериментаторы» получат возможность стажировки в США или гранты на дальнейшие исследования.

Королевская PиRушка

С 14 по 17 сентября в выставочном центре *КиевЭкспоПлаза* состоялась *восьмая Международная выставка рекламы, маркетинговых технологий и масс-медиа* **REX-2004**, проводимая компанией **EuroIndex**. Среди участников выставки было более 340 рекламных агентств, производителей рекламы, газет и журналов, телерадиокомпаний, а также студии графического дизайна, web-дизайна и многие другие.

Выставка была разделена на три тематические группы: *рекламно-сувенирная продукция и промоушн-материалы; дорожная реклама, материалы и технологии для ее изготовления; графические системы и цифровая печать.*

Устроители выставки решили немного расширить традиционные рамки, и в этом году были представлены и другие интересные направления; также функционировал консультационный PR-центр и консультационный центр *Маркетинговые исследования.*

Среди большого количества профессиональных конференций, мастер-классов и презентаций в программе этого года следует особо отметить такие мероприятия как *День PR-технологий, День медиа* и *День маркетинговых исследований.*

В целом, REX-2004 удалась затронуть и осветить наиболее актуальные вопросы маркетинга и рекламы, а также современные технологии, используемые в этих областях.

Одомашненный FineReader

Компания **ABBYY Украина** объявляет о выходе нового программного продукта **ABBYY FineReader 7.0 Home Edition**, который представляет собой специальную, «облегченную» версию настольной системы распознавания символов *ABBYY FineReader 7.0* и предлагает самые необходимые для домашних пользователей функциональные возможности для эффективной обработки документов.

Согласно исследованиям компании ABBYY при выборе программного продукта по преобразованию документов в электронный вид на дому, решающими факторами являются доступная стоимость решения, высокая точность распознавания и удобство использования. Все эти качества реализованы в новой версии ABBYY FineReader 7.0 Home Edition.

Основные функциональные возможности ABBYY FineReader 7.0 Home Edition таковы:

✓ сканирование и распознавание по нажатии одной кнопки;

✓ распознавание смешанных многоязычных текстов на 177 языках;

✓ распознавание многостраничных документов;

✓ сохранение структуры документа с учетом компановки, колонок, рисунков, таблиц;

✓ пошаговое сохранение результата распознавания с помощью «Мастера сохранения»;

✓ сохранение документов в формате PDF;

✓ сохранение результатов распознавания в форматах RTF, DOC, XLS, HTML и TXT;

✓ прямой экспорт результатов распознавания в Word, Excel, Lotus Word Pro, WordPerect, Adobe Acrobat.

Как отметил *Виталий Тищенко*, директор ABBYY Украина, выпуск ABBYY FineReader 7.0 Home Edition является продолжением продуктовой стратегии ABBYY, которая заключается в стремлении обеспечить различные аудитории пользователей доступным и качественным ПО. Теперь система ABBYY FineReader 7.0 нашла и себя и в сегменте домашних пользователей. Продукт поступит в продажу в начале октября 2004 года.

Также ABBYY Украина объявила о выходе совершенно нового программного продукта — ABBYY PDF Transformer 1.0, который позволяет преобразовывать любые типы PDF-файлов в привычные для редактирования форматы, такие как Microsoft Word, Microsoft Excel, HTML и TXT. При этом обрабатываемые документы могут содержать таблицы, картинки, колонки и заголовки — ABBYY PDF Transformer 1.0 сохранит оформление исходного документа и после преобразования предложит полноценную «копию» в нужном формате.

Работа с ABBYY PDF Transformer 1.0 не требует наличия специальных технических знаний. Продукт довольно прост и понятен в использовании — для преобразования нужной информации достаточно нескольких щелчков мыши. Таким образом, применить решение в своей работе смогут специалисты различных сфер деятельности. Подробнее о новинке мы расскажем в одном из ближайших номеров «МК».

Две звезды на погонах «РФ»

С 16 по 19 сентября в Харькове уже в шестой раз прошел фестиваль **Звездный мост**, одно из главных событий года в жизни фэндома на всем постсоветском пространстве. В этом году в фестивале приняли участие такие известные фантасты, как С. Лукьяненко, В. Головачев, О. Дивов, А. Громов, А. Орлов, В. Панов, М. Дяченко, В. Аренев, П. Верещагин, К. Якименко, В. Былинский, В. Васильев, Р. Злотников, Ю. Брайдер и Н. Чадович, Ю. Буркин, Г. Л. Олди, А. Валентинов, А. Бессонов, А. Зорич и многие другие. Также приехали представители издательств *Solaris* (Польша), *Эксмо-Пресс, Альфа-Книга* (Москва), *Зеленый Пес* (Киев), *Сталкер* (Донецк), *Азбука, Домино, Северо-Запад, Астрель-Спб* (СПб), сотрудники журналов и газет *Реальность фантастики, Интересная газета* (*Просто фантастика*) (Киев), *Если* (Москва), *Порог* (Кировоград), *Питер-book* (Санкт-Петербург), *Meridian* (Ганновер, Германия), известные критики, литературоведы и переводчики из Украины, Польши, России, Беларуси, а также многие менее известные писатели, художники, книготорговцы и просто любители фантастики. Были аккредитованы корреспонденты многих газет, журналов, теле- и радиоканалов Украины и России. (Всего официально зарегистрировались более 300 участников фестиваля).

Помимо встреч с читателями в библиотеках, клубах и книжных магазинах, в рамках фестиваля зачитывались доклады о состоянии современной украинской фантастики, прошел традиционный семинар молодых авторов, работали философская секция, секция толкиенистики, фантастического оружия, творческой мастерской *Второй блин* и т.д., состоялись поэтические встречи, художественные выставки, игра «Что? Где? Когда?»; рыцарский турнир, чемпионат по пейнтболу «НФ против фэнтези», концерт и праздничный фейерверк.

Наш журнал **Реальность фантастики**, по итогам «Звездного моста», стал лауреатом сразу двух премий: премии творческой мастерской *«Второй блин»* — за стирание граней между реальностью и фантастикой, а также премии *Шпага братства* — за популяризацию нашей фантастики в Германии. Поздравляем!

Ну а подробный отчет о фестивале читайте на страницах октябрьского номера «РФ».

ИГРОВЫЕ НОВОСТИ

Возвращение во Вьетнам

Компания **«Новый Диск»** объявила о приобретении прав на локализацию и издание на территории стран СНГ 3D-шутера **ShellShock**, созданного **студией Guerilla Games**. Русский релиз игры должен состояться двадцать третьего сентября этого года. Большинство из вас



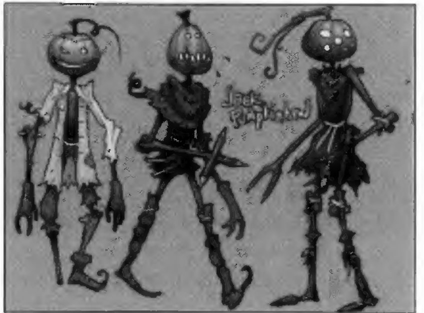
наверняка слышало об этой игре. Разработчики грозились показать нам все ужасы войны во Вьетнаме и поразить игроков невиданной жестокостью и реализмом. К сожалению, исполнить свои обещания им не удалось. ShellShock представляет собой вполне стандартный аркадный экшен. Нам снова предстоит выполнять вполне стандартные миссии по уничтожению вьетконговцев и спасению заложников, по поиску складов оружия и захвату стратегически важных точек. Впрочем, поклонники игры «про Вьетнам» вполне могут провести за игрой пару-трайку увлекательных вечеров.

Новые приключения «Красной Акулы»

Компании **ИДДК** и **G5 Software** анонсировали разработку игры **«Красная Акула 2»**, которая, как нетрудно догадаться из названия, представляет собой продолжение вертолетной аркады «Красная Акула», увидевшей свет в 2002 году. Сюжет первой части игры был закручен вокруг некоего фантастического эксперимента, в рамках которого российский боевой вертолет Ка-50 «Красная Акула» был отправлен во времена Великой Отечественной войны на помощь союзным войскам в борьбе с фашистами, и изменил не только ход основных сражений второй мировой, но и судьбы участников этих событий. Сюжет «Красной Акулы 2» напрямую связан с событиями, разворачивавшимися в первой части игры. «Ганс Хоссманн, молодой ученый-физик третьего рейха, в детстве стал свидетелем одного из этапов коррекции истории. На всю жизнь в его памяти остались воспоминания о грозном посланце из будущего. В послевоенные годы Хассманн сумел разработать собственную конструкцию машины времени. Вернувшись с ее помощью во времена гитлеровской Германии, он направил свой гений на создание фантастических для того времени видов вооружения, чтобы изменить исход второй мировой войны. Планетарная антитеррористическая организация будущего, в ведении которой к 2013 году находится технология путешествия во времени, узнает о коварных планах Хассманна и начинает очередную операцию по коррекции истории в надежде помешать ему изменить ход событий. В новой игре вам предстоит стать пилотом боевого вертолета и принять участие в самых сложных этапах этой операции. Вас ожидают масштабные воздушные бои, поддержка сухопутных и морских операций, спасательные и разведывательные миссии, а также участие в легендарных сражениях второй мировой». Действие игры разворачивается в Европе, в тихоокеанском и средиземноморском регионах. Для выполнения боевых задач игроку предоставляются три современных боевых вертолета: *Ка-50 Hokum, AH-64A Apache, PAH-2 Tiger*. Игра разрабатывается на модифицированном движке «Красной Акулы», и должна появиться в продаже уже в первом квартале следующего года. Если вы хотите узнать больше о данном проекте, обязательно загляните на недавно открывшийся официальный сайт игры (**http://www.redshark.g5software.ru**).

Дороти остается в Канзасе

Компания **Carbon 6**, возглавляемая небезызвестным **American`ом McGee**, на днях официально заявила об остановке работ над проектом American McGee`s Oz. Согласно новому договору, заключенному МакГи с киностудиями *Walt Disney Pictures* и *Jerry Brukheimer Films*, игра должна появиться в продаже одновременно с художественным фильмом, к съемкам которого вышеозванные студии собираются приступить



в самое ближайшее время. Соответственно, задерживается выход книги от *Warner Books* и настольной ролевой игры. К сожалению, на сегодняшний день не известно, когда фильм должен выйти на экраны, так что поклонникам таланта господина МакГи, похоже, придется запастись терпением.

Сайт Блицкрига

Компания **Nival Interactive** объявила об открытии официального сайта исторической стратегии **«Блицкриг 2»**. Так что, если вы интересуетесь данным проектом, заходите на **http://www.nival.com/blitzkrieg2_ru** и читайте самую свежую и достоверную информацию. Как большинство из вас, конечно, знает, «Блицкриг 2» является продолжением революционной стратегии «Блицкриг», действие которой разворачивается во времена второй мировой войны. Благодаря уникальному генератору миссий, созданному специалистами «Нивала», вы сможете выбирать именна те миссии, которые интересуют вас больше всего. Глобальные сражения танковых армий, операции в тылу врага, молниеносные наступления или оборона городов — любой из вариантов вы без труда найдете в этой игре. Обязательными для прохождения являются лишь немногочисленные сюжетные миссии, а каким образом вы подготовите свое подразделение к «сюжетным» сражениям — зависит только от вас. «Блицкриг 2» будет выгодно отличаться от своего предшественника новым графическим движком, именуемым *Enigma Engine*, который поддерживает самые современные эффекты. Разработчики обещают нам небывалую интерактивность окружающего мира, широкие возможности при стратегическом планировании операций и, конечно же, множество разнообразных миссий, среди которых мы сможем выбирать именно то, что нам больше всего по душе. Не забыт и мультиплейер. В «Блицкриг 2» планируется сетевая игра по локальной сети и через специальный турнирный сервер. Релиз игры намечен на первый квартал 2005 года. Ждем с нетерпением.

Второе пришествие Барда

Компании **THQ** и **InXile Entertainment** объявили о подписании договора на издание мультиплатформенной ролевой игры **The Bard`s Tale**, разрабатываемой одновременно для платформ *PC, PlayStation 2* и *X-box*. Данный проект является римейком старой игры The Bard`s Tale, вышедшей в 1985 году на *Apple II*. На этот раз нам предложат роль беспринципного наемника, одинаково хорошо владеющего как мечом, так и заклинаниями при-



зыва (суммонинг), который ради денег готов влезть в любую передрягу. Разработчики обещают нам нелинейный сюжет — причем ветвление возможна как при выборе определенных фраз в диалогах, так и в зависимости от действий главного героя — и общую юмористическую направленность игры. По сути, в The Bard`s Tale мы найдем множество моментов, пародирующих наиболее распространенные RPG`шные штампы. Игра разрабатывается на движке довольна популярной консольной action/RPG *Baldur`s Gate: Dark Alliance*, а возглавляет команду разработчиков небезызвестный **Брайан Фарго**, создатель ныне покойного *Interplay*. Версии The Bard`s Tale для PlayStation 2 и X-box должны появиться в продаже уже второго ноября этого года, а PC-релиз намечен аж на следующую весну.

Социальный взлом

Сергей ПАРИЖСКИЙ

Часто хакеров считают высшими существами, которые способны взломать все, что угодно. Для достижения своей цели они, по общему мнению, в течение многих ночей с 24:00 до 4:00 изучают защиту жертвы, пишут километры кода и выполняют еще много рутинной работы, сидя в темном терминале UNIX'а. А на самом деле иногда для взлома хватает одного письма на E-mail. Такие методы взлома называются социальной инженерией, а как они проводятся, и как не стать их жертвой, вы узнаете из этой статьи.

Метод №1: Администрация

Первый, самый популярный, метод социальной инженерии — это представиться вышестоящим органом и убедить пользователя, что вы имеете полное право на интересующую вас информацию. Например, вы — пользователь, и адрес вашей почты **user@mail.ru**. В один прекрасный день вы получаете письмо от **administratar-@mail.ru** с темой «ВАЖНО!» и таким содержанием:

«Здравствуйте, User! В последнее время увеличилось количество жалоб на то, что через наш smtp-сервер проходит большое количество спама. Чтобы отделить легитимных пользователей нашей почтовой службы от спамерских роботов, мы начали проводить подобные проверки. Все, что нужно для подтверждения своей личности, это отправить нам свой пароль. В противном случае ваша учетная запись будет удалена. Заранее благодарим за сотрудничество.

С уважением, администратор почты @mail.ru».

Как вы понимаете, это письмо было написана не администратором, а человеком, который захотел получить пароль на ящик **user@mail.ru**. Должен сказать, что немало людей отошлет свой пароль в ответ на такое письма. Люди, каторые отсылают пароль па такой прасьбе, часто допускают еще адну ашибку — используют один пароль на все случаи жизни. А значит, автор письма одним махам забирает E-mail, ICQ (если есть) и dial-up в придачу.

Меры противодействия

Человек, писавший эта письмо, провел небольшую подгатавку. Во-первых, создал e-mail на том же сервере, чта и пользователь. Ва-вторых, написал письмо настаящему администратору, чтобы узнать, каков стиль написания писем и как они падписываются. Последнее, что астается сделать, — эта сочинить письмо с просьбай выслать пароль.

Бальшая часть пальзователей после того, как они утратят свой ящик, даже не заподозрят фиктивного администратора в пропаже параля. Другими словами, не будут учиться на своих ошибках. Закономерность, что после писем «администратара» пропадает ящик, они найдут толька пасле письма шестого.

Запомните, ни при каких обстоятельствах ни администратар, ни служба паддержки почты никогда не попрасит прислать им ваш пароль! Администратор сам может сделать все с вашей учетнай записью без вашега разрешения. Также не попадайтесь на удочку, когда вас прасят поменять параль на указанный в письме. Па сути, это то же самое, что назвать им свой пароль.

Метод №2: По дружбе

Представьте ситуацию: вы с другом занимаетесь паддержкой какаго-то сайта, и оба имеете к нему доступ по ftp. Однажды вы балтаете с другом по ICQ, речь заходит а вашем сайте, он говорит, что нашел на сайте один баг, каторый надо срачна исправить, а пароль к ftp он забыл. Так как сайт общий, и друг имеет на нега такое же право, как и вы, то, канечно, вы атошлете ему параль. Не удивляйтесь, если через считанные минуты на индексной странице будет красоваться *«DeFaCeD by $UPER_X@KER»*.

Меры противодействия

Взлом номера ICQ сегодня не считается чем-то ачень сложным. Существует многа праграмм, катарые делают это локально и удаленна. Подготовка не заняла мнаго времени: сначала взломщик посетил сайт друзей и узнал их почтовые адреса, благодаря чему и ICQ (если ани не были указаны на сайте). Затем ан некотарое время переписывался с одним из друзей, чтобы узнать его (на случай некоторых идентифицирующих вопросов), прочитал его данные, указанные в ICQ и, наконец, изучил раздел «About» на их сайте. После этаго последовал взлом ICQ и небальшай диалог от имени аднаго из товарищей с выяснением пароля.

Запомните — до тех пор, пока вы абсалютно не будете уверены, с кем общаетесь, будьте крайне сдержанны в своих словах. Если вы хатите быть в безопаснасти, надо всегда быть немнога параноикам. Прежде чем сообщать какие-либо приватные данные, надо правести идентификацию, задав ряд вопросов. Для начала можно узнать, помнит ли друг длину параля или какую-либо его часть. Затем узнайте, что именно он хотел исправить на сайте, после чего уверьте, что сами сейчас все исправите, а параль вы ему скажете при личной встрече. Наблюдайте, наскалько быстро он отвечает на вопросы, но помните, что взломщик готов к вашему дапросу. Подключившись к вашему другу по ICQ пад видом девушки, он мог выведать очень многое. Так что задавайте больше вопросов. И не следует пасле того, как вы узнали, что он действительна учился в шкале №109, отправлять ему пароль и одалживать да завтра деньги по Web-Money ☺.

Метод №3: Юный хацкер

Когда вы заходите в какой-либа хакерский чат, хакерский канал IRC или просто общаетесь с человеком из такого круга по ICQ, абычно все хвалятся своими взломами и разными хаками. У вас невольно возникает желание и самому что-то взломать, но не хватает знаний. Вдруг один из ваших «продвинутых» собеседников говарит, что готав поделиться секретами взлома. И савершенно бесплатно может научить взламывать любой ящик на **yandex.ru**. А секрет такой:

*Когда вы забыли свой пароль, Яндекс предлагает воспользоваться функцией «Напомнить пароль». При нажатии на эту ссылку вы попадаете на страницу, где надо будет заполнить данные, которые вы вводили при регистрации (дата рождения, имя...) и, конечно же, ответить на секретный вопрос. Если все данные будут правильными, то пароль будет отослан на указанный e-mail. Секрет в том, что данные обрабатываются mail-роботом (**mail_robot@yandex.ru**), а его можно обмануть. Для этого надо со своего ящика отправить пустое письмо, а в поле «тема» написать строку:*

$$login_жертвы$$ваш_login$$ваш_пароль$$;

Для пользователя **user@yandex.ru** с паролем 12345, который хочет поломать ящик **lamer@yandex.ru**, поле Темы должно выглядеть так:

$$lamer$$user$$12345$$;

Если пароль правильный, то вскоре вы получите письмо от **mail_robot@yandex.ru**, где будет пароль к ящику **lamer@yandex.ru**.

Меры противодействия

Подготовка не заняла много времени. Нужно было создать ящик **mail_robot@yandex.ru** и сочинить историю, чтобы пользователь прислал на новый ящик пароль. Здесь явно напрашивается фраза: «Не рой другому яму — сам в нее попадешь». Взлом чужой почты — это не есть гуд, так что люди, которые идут на это, савершенно справедливо заслуживают, чтобы с ними так паступали. При попадании в такую ситуацию можно посоветовать — не будьте сталь доверчивыми. Не думайте, что из тумана появится челавек, который будет учить вас хитрастям и уловкам хакеров. В данном случае разумно было создать одноразовый ящик, а не использовать свой.

Также ваш новый «друг» может прислать вам программу для взлома ящиков. Будьте уверены, чта при помощи этай программы вы не только не взламаете чужой ящик, а еще и лишитесь своего.

Метод №4: Внимание! Конкурс!

Все мы любим получать подарки, выигрывать призы в канкурсах, и тому подобные приятные шары. На этом и основывается данный метад обмана. Вам могут написать письма примерно следующего содержания:

«Внимание!!! Конкурс с призами!

Обьявляется конкурс на самый сложный и оригинальный пароль.

Для того что принять участие не надо проходить никаких регистраций.

Все что нужно это прислать наш вариант самого-самого пароля.

Одно усолвие: этот пароль должен быть обьязательно использован вами.

Итоги подводятся в конце месяца, победитель получает 100 $.

В случае вашей победы мы вам напишем.

С уважением, Very Good Company».

Меры противодействия

В большинстве случаев — это липовые конкурсы, которые проводятся хакерами, а вы — единственный участник ☺. Но если вы оптимист, и у вас есть желание поучаствовать, то вы должны проверить три момента, которые должны вас настарожить. Ва-первых, нет ли в письме грамматических и орфографических ошибак. Во-втарых, не используют ли эти люди, гатовые платить $100, бесплатный e-mail. И, наконец, имеется ли в конце письма падпись с указанием названия компании. Даже если все правильно, и ни к чему не придерешься, все-таки создайте одноразовый ящик и установите на него одноразовый пароль.

Метод №5: Распродажа

Представьте ситуацию: к вам приходит прайс от предприятия, которое проводит распродажу. Вы уже хотите заносить их адрес в черный список, но тут замечаете цены, по которым они прадают товары. Сначала вы думаете, что это опечатка, — но, прочитав дальше, понимаете, что сам Бог прислал вам это письма, так как все, что вы хотели, но из-за высокой цены не могли купить, продается здесь со скидками 50% и более. Единственное, чта надо сделать, это пройти регистрацию на сайте продавца. Затем можно покупать все, что угодно. Для регистрации нужно заполнить стандартную форму: указать пароль для своей учетной записи и все свои данные.

Меры противодействия

Как вы понимаете, хакеры ничега не продают, тем более, по заниженным ценам. Все введенные вами данные будут записаны скриптом в блакнот. Здесь делается ставка на то, что вы используете один параль и для регистрации на сайте, и для своей почты. При неудаче у хакера астается надежда воспользоваться функцией «Напомнить пароль». Если вы ввели адинаковые данные при регистрации ящика и при регистрации на сайте «прадавца» (что логична), то пароль будет атослан хакеру.



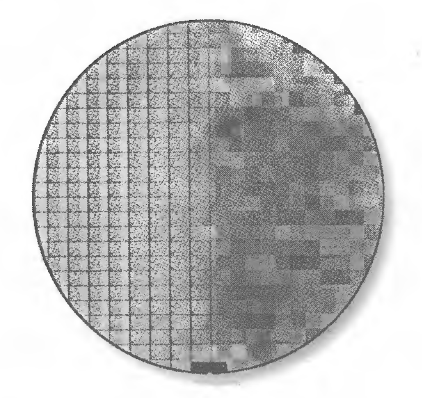
Серенада Силиконовой Долины

Сергей Н. МИШКО
maestro@mycomputer.ua

За окнами осень — пора постепенного увядания природы, но одновременно и пора оживления бизнеса после летнего затишья индустрии. Хотя для компании Intel (www.intel.ru) летний период трудно было назвать затишьем, уж слишком много принципиально новых продуктов появилось на рынке благодаря ее стараниям. Американская сессия осеннего IDF (Intel Developer Forum) подвела некоторый промежуточный итог деятельности корпорации и наметила новые перспективы для отрасли. Автор материала уже в третий раз посещает этот форум и спешит в очередной раз поделиться своими впечатлениями и полученной информацией.

Как всегда, краткая справка для тех, кто впервые слышит об **IDF**. Своим существованием данный форум обязан компании **Intel**, которая выступает его организатором. IDF состоит из нескольких конференций, которые проходят в различных точках мира. Серии конференций составляют сессии форума — весеннюю и осеннюю. Традиционно очередную сессию открывает американская конференция. Первая сессия IDF прошла весной 1997 года — таким образом, осенью этого года в США открылась уже 16-я по счету сессия форума.

главного директора по технологиям (Senior Vice President and CTO — Chief Technology Officer). Он ознакомил присутствующих с программой форума и объявил о местах проведения предстоящих конференций IDF в рамках осенней сессии. Впервые за всю историю форума этой осенью он пройдет также в Южной Америке. В Москве осенний IDF будет проходить уже третий год подряд (см. статьи Сергея Н. МИШКО и Владимира СИРОТЫ «Москва встречает IDF», МК, №41 (212), и «IDF в Белокаменной», МК, №46, 48 (269, 271)).



Intel Developer
FORUM

Местом проведения американской конференции, как и предыдущей, весенней (см. статью Сергея Н. МИШКО «Технологии из-за океана», МК, №10–14 (285–289)), стал Сан-Франциско, штат Калифорния. Хотя, например, в прошлом году IDF проходил в другом калифорнийском городе Сан-Хосе (см. статьи Сергея Н. МИШКО «Однажды в Америке», МК, №11–12 (234–235), и COMPOSTER'а «Осенний IDF после жаркого лета...», МК, №41–42 (264–265)). В любом случае, оба города находятся в самом центре знаменитой Силиконовой Долины — средоточия мира информационных технологий.

Нынешний IDF прошел под девизом *«Stretching Boundaries — Building Relationships»*, что в переводе означает *«Сотрудничество без преград»*. Этот лозунг хорошо передает идею самого мероприятия, которая состоит в привлечении ключевых игроков рынка к взаимодействию, совместной разработке стандартов и их претворению в жизнь. Подтверждением тому служит число посетителей — их общее количество на всех конференциях весенней сессии этого года составило около 17 000 разработчиков, инженеров, представителей прессы.

Если весенняя сессия американского IDF проходила в здании Moscone Center West, то осенняя прошла в соседнем здании Moscone Center South. Форум продолжался в период с 7 по 9 сентября, после чего представители прессы имели возможность посетить дополнительную сессию по бенчмаркингу и съездить на завод Intel, расположенный в Санта-Кларе, Калифорния.

More than MHz

Осенний IDF'04 начался с приветствия Патрика Гелсингера (Patrick Gelsinger), старшего вице-президента Intel и

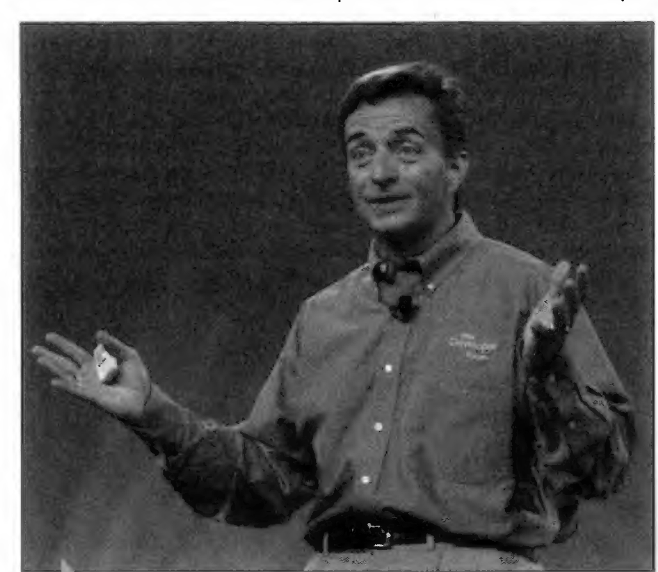
Сразу за вступительной речью Патрика Гелсингера последовало выступление Пола Отелини (Paul Otellini), президента и главного директора по операциям (President and COO — Chief Operating Officer). Он отметил, что несмотря на повсеместный спад в индустрии, произошедший в 2000 году, в последние годы снова наблюдается рост всех сегментов рынка.



Регистрация участников IDF в лобби Moscone Center South

Конвергенция, о которой так много говорили в последнее время, стала реальностью, цифровой стиль жизни становится все более обыденным, и в перспективе Intel рассчитывает на 3 млрд. новых пользователей по всему миру.

Проникновение цифровых технологий в самые различные отрасли означает необходимость не только наращивания вычислительной мощности отдельных компонент, но и добавления в вычислительные системы различных технологий, востре-



Патрик Гелсингер во время открытия IDF

бованных той или иной группой пользователей. Другими словами, Intel будет делать акцент на платформы. За примерами далеко ходить не нужно — достаточно вспомнить прекрасно себя зарекомендовавшую технологию *Centrino* для мобильных ПК и набирающую обороты *The Digital Home Platform*. В основе любой платформы лежат определенные ключевые технологии, в разработке которых Intel тоже принимает самое непосредственное участие. Подробнее обо всем этом в соответствующих разделах.

Одним из ключевых моментов стратегии Intel Пол Отелини назвал развитие широкополосного беспроводного доступа. Речь идет о дальнейшем совершенствовании стандарта *Wi-Fi* и скорейшем внедрении *WiMAX*. В подтверждение слов присутствующим продемонстрировали первую интегрированную систему стандарта 802.16-2004 на силиконовом чипе. Подробнее тему WiMAX мы затронем в разделе, посвященном беспроводным коммуникациям.

Вторым ключевым моментом, на котором Intel акцентирует особое внимание, является параллелизм вычислений. Если первым шагом на пути к его реализации стала представленная в ноябре 2002 года технология *Hyper-Threading* (см. статью Владимира СИРОТЫ «Бурный поток вычислений», МК, №48 (219)), очередным этапом развития концепции станет использование многоядерности во всех (!) линейках процессоров уже в следующем году. Детальнее о самих процессорах читайте в тематических разделах.

The T's

Прежде, чем перейти к обзору определенных направлений деятельности Intel, остановимся на наиболее важных, с точки зрения компании, технологиях, которые она использует или будет использовать в своих процессорах и наборах системной логики. Не считая Centrino — ей мы посвятим отдельный раздел — их пять. Детальнее о некоторых из них можно прочесть в статье Владимира СИРОТЫ «Технопарty IDF'a» (МК, №1, 3 (276, 278)). Начнем с уже реально доступных и работающих технологий.

✓ *Hyper-Threading Technology (HT)*

Эта технология является самой «старой» из всех, которые мы перечислим в этом разделе. Она уже скоро два года, как находит себе успешное применение в настольных процессорах Intel Pentium 4. Еще раньше она появилась в процессорах Intel Xeon. На сегодняшний день все серверы и 55% клиенских систем на основе архитектуры Intel поддерживают HT.

✓ *Extended Memory 64 Technology (EM64T)*

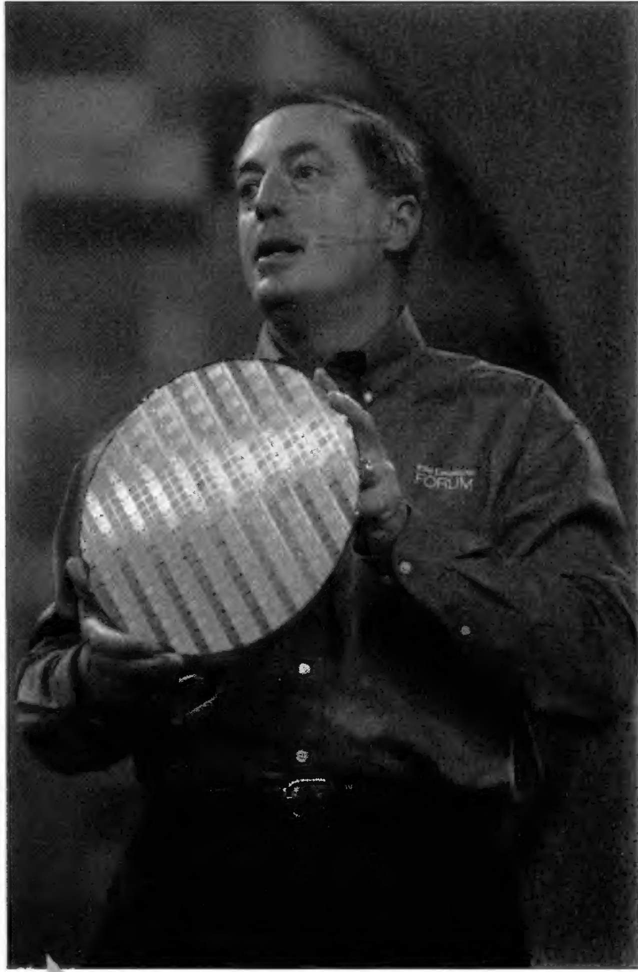
Технология, позволяющая 32-разрядным процессорам выполнять 64-разрядные инструкции и, таким образом, работать в 64-разрядных средах. Анонсированная на весеннем IDF'04, сейчас она доступна в процессорах Xeon для серверов и рабочих станций, работающих под управлением 64-разрядных версий Windows и Linux.

✓ *LaGrande Technology (Security, LT)*

Прерогативой этой технологии является защита информации, снижение степени риска атаки на систему, распределение прав доступа и т.п. Ее появление ожидается только с выходом новой версии Windows, известной под кодовым названием *Longhorn*.

✓ *Vanderpool and Silvervale Technology (Virtualization, VT/ST)*

Настольный и серверный вариант технологий виртуализации, их поддержка тоже появится с выходом Longhorn. Идея этих технологий чрезвычайно интересна — они позволяют на базе одной системы создавать независимые друг от друга разделы и устанавливать на них совершенно разные операционные системы и приложения. Во время своего выступления Пол Отелини продемонстрировал реально работающую систему, состоящую из четырех отдельных разделов. На двух из них выполнялись корпоративные и персональные приложения под управлением Windows, на третьем — инженерные приложения под Linux и на последнем — программы управления IT-инфраструктурой.



Пол Отелини демонстрирует кремниевую пластину с чипами, изготовленными по нормам 65-нм техпроцесса

✓ *Active Management Technology (AMT)*

Подсчитано, что в настоящий момент на поддержку IT-инфраструктуры бизнеса тратится 80% средств, тогда как на модернизацию — только оставшиеся 20%. Технология AMT, о которой Intel объявила на этом IDF, призвана упростить поддержку парка ПК, использующего самые разные платформы — от карманных коммуникационных устройств до серверов. Ее первые спецификации компания пообещала объявить на следующем весеннем IDF'05.

Desktop Platforms

Настольная платформа Intel за полгода, прошедшие с момента проведения весенней сессии IDF'04, претерпела, мож-

но сказать, революционные изменения. С появлением новых чипсетов i915P/G, i925X Express она обрела поддержку процессорного разъема LGA775, шины PCI Express (см. статью Александра ВОЛОХИ «Expressивная шина», МК, №47, 51 (270, 274)), памяти нового поколения DDR2, интегрированного видео, совместимого с DirectX 9 (в чипсете i915G), 8-канального High Definition Audio и ряда других полезных функций.

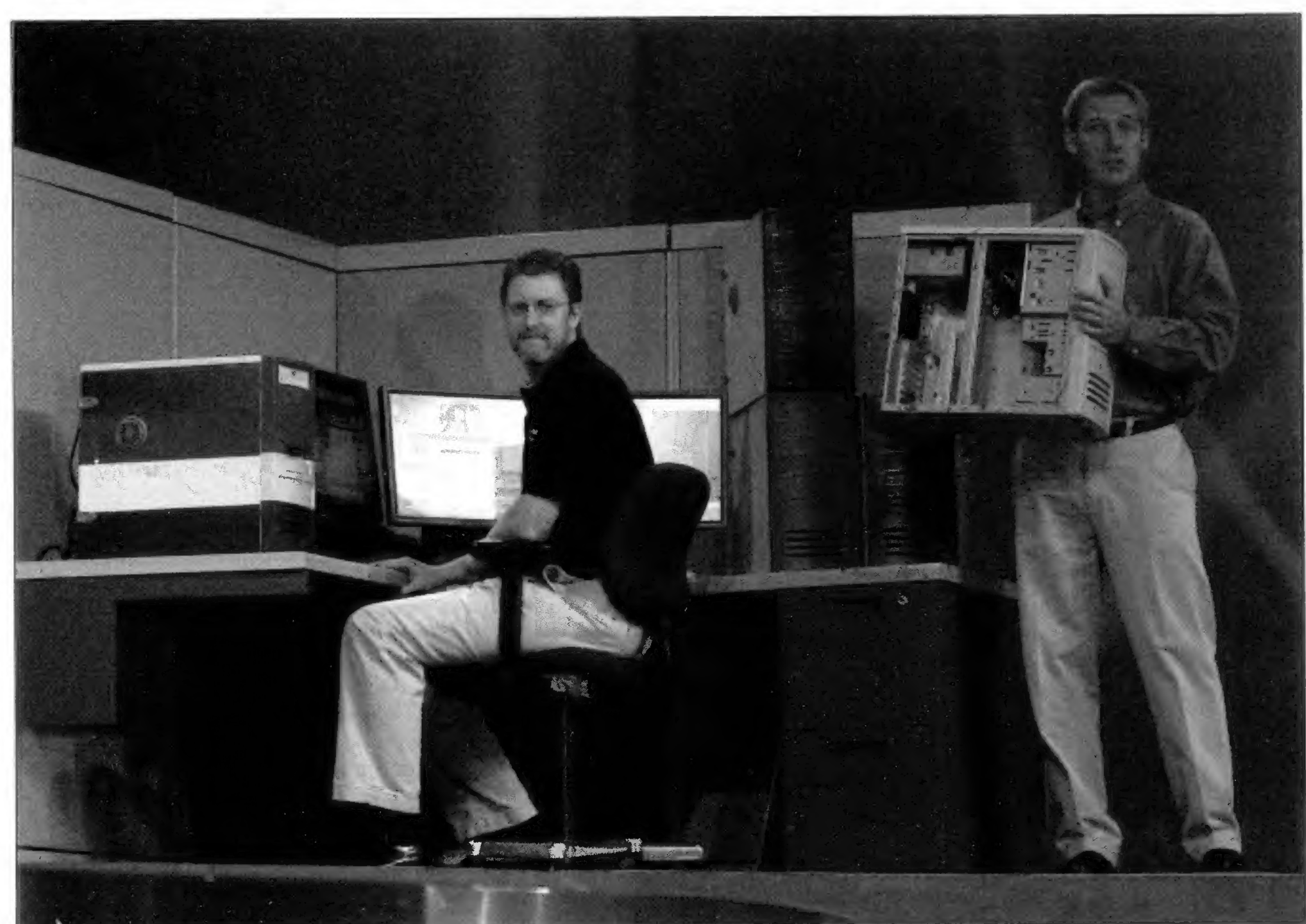
Продолжает свое развитие и линейка процессоров Pentium 4 на новом ядре Presscott, изготавливаемом по нормам 90-нм техпроцесса. В настоящий момент они имеют кэш второго уровня объемом 1 Мб и системную шину 800 МГц. Тактовая частота самой старшей модели Pentium 4 560 составляет 3.6 ГГц. По словам Отелини, Intel в перспективе увеличит кэш второго уровня процессоров Pentium 4 до 2 Мб и поднимет частоту системной шины до 1066 МГц. Очевидно, претерпят похожие изменения и характеристики облегченных Intel Celeron.

На второй день IDF Билл Сью (Bill Siu), вице-президент и генеральный менеджер (Vice President and General Manager) подразделения Desktop Platforms Group, продемонстрировал в работе двухъядерный настольный процессор с поддержкой технологии HT. Напомним, его появление ожидается в следующем году с внедрением нового 65-нм техпроцесса (см. статью Сергея Н. МИШКО «Утонченный чип», МК, №36 (311)). На демонстрации можно было наблюдать в системе 4 виртуальных процессора. По прогнозам Intel в 2006 году почти половина всех ее настольных процессоров станут двухъядерными.

Не менее важным фактом является и продолжение сотрудничества Intel с *Dolby Laboratories* в рамках инициативы Dolby PC Entertainment Experience, о которой компании впервые заявили на весеннем IDF'04. Основным элементом этого взаимодействия является программа Dolby PC Logo, призванная помочь покупателю выбрать наиболее подходящий для его нужд ПК на базе платформы Intel. Результатом ее работы стали 3 логотипа, представленные на нынешнем IDF, — *Sound Room, Home Theater и Master Studio.* Теперь компьютеры на основе чипсетов i915, i925X Express смогут проходить сертификацию, о чем будет свидетельствовать наличие того или иного логотипа Dolby.

(Продолжение следует)

Билл Сью говорит о двухъядерных процессорах



Демонстрация технологии Vanderpool (VT)

На витрине: системная плата AOpen i915Pa-E

Александр КОНДАУРОВ
руководитель отдела научно-технической информации K-Trade

По традиции осень — это пора «урожая» новинок, анонсированных летом, но до витрин добравшихся к пику «горячего сезона». Сегодняшняя новинка — системная плата **AOpen i915Pa-E (рис. 1).**

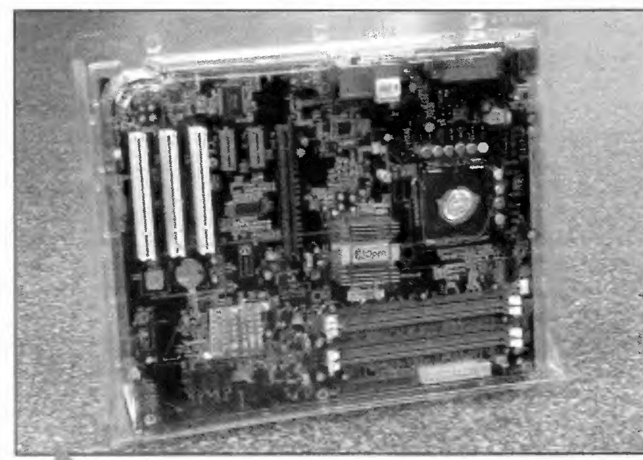


Рис.1

Знатоки и просто те, кто следит за рынком, наверняка заметили, что с приходом новых чипсетов, шин и технологий *AOpen* сменила и систему маркировки плат, сделав ее значительно более информативной. Прежние условные номера моделей заменены теперь на стройную систему букв и цифр, напрямую указывающих на основные характеристики системной платы. Так, AOpen i915Pa-E расшифровывается как плата производства AOpen под платформу *Intel* (**i**), построенная на чипсете **i915P**, формата *ATX* (**a**, **m** означает microATX), оборудована *PCI-Express Gigabit Ethernet* (**E**). С учетом параметров, обеспечиваемых чипсетом, теперь при виде названия платы можно практически полностью представить ее возможности.

На сей раз «образ», сформировавшийся при разборе названия платы, оказался если в чем-то и не точным, то только в том, что отдельные детали платы превзошли ожидания.

Началось все с упаковки. Кроме традиционного прозрачного окна в коробке над платой, позволяющего увидеть основные узлы платы (**рис. 2**), предусмотрено еще одно — напротив блока разъемов задней панели, чтобы покупатель сразу заметил два последовательных порта (согласитесь, не очень частая в наши дни роскошь) и увеличенный блок аудиоконтактов. Несмотря на широко пропагандируемую новую звуковую технологию *Azalia*, поддерживающую 7.1-канальный выход, большинство производителей плат на i915 пока предпочитают обойтись более простым, 5.1-канальным кодеком *Realtek*

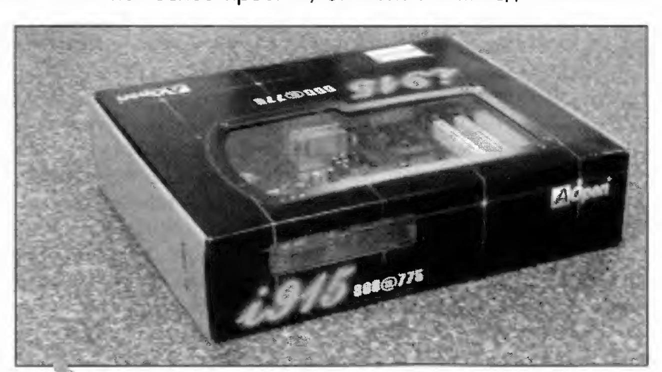


Рис.2

ALC860. AOpen же выбрала для себя стандартным кодек **Realtek ALC880**, полностью реализующий все возможности Azalia, включая 7.1-канальный аудиовыход и Dolby Digital Surround EX.

Как и большинство плат на i915, AOpen i915Pa-E предназначена для работы с процессорами, устанавливаемыми в сокет *LGA775*, отличающийся от всех предыдущих в первую очередь тем, что «ножки», а вернее, пружинные контакты, теперь расположены не на процессоре, а на сокете. Что, во-первых, снижает риск повреждения ножек при непрофессиональном обращении. А, во-вторых, выглядит экономически оправданным: современный процессор стоит в несколько раз дороже системной платы. Фирменный «четырехтактный» преобразователь питания позволяет использовать на этой плате все существующие на сегодняшний день процессоры, устанавливаемые в сокет LGA775.

Идя навстречу пожеланиям рынка, пока не готового полностью перейти на новую (и, соответственно, заметно более дорогую) память *DDR2*, AOpen предлагает использовать на плате до четырех модулей привычной памяти **DDR400**, организованных в двухканальную схему (собственно, при системной шине 800 МГц, обеспечиваемой чипсетом, такая память может оказаться даже быстрее DDR2 варианта).

А вот с видеокартой привычными традиционными *AGP* решениями обойтись уже не удастся — для ее установки на плате есть слот **PCI Express x16**, так что желающим получить высокопроизводительную графическую систему придется отдать дань прогрессу и купить соответствующую видеокарту. Те же, кто собирается использовать плату с видеосистемой минимальной навороченности, могут ограничиться какой-нибудь старой картой, устанавливаемой в PCI-слот. На плате таких три, так что места хватит.

Кроме упомянутых слотов PCI, для расширения функциональности платы можно воспользоваться еще двумя слотами **PCI Express x1**. Например, в них можно установить вторую сетевую карту или модем. Основной же Ethernet-контроллер, базирующийся на чипе *Marvell 88E8053*, подключен к еще одному каналу PCI Express, и благодаря этому может, не мешая другим устройствам, обеспечивать гигабитный поток сетевых данных.

Еще одно «смещение акцентов», по сравнению с традиционным вариантом, произошло в организации дисковой подсистемы. Казалось бы, незыблемое правило «IDE каналов на любой системной плате всегда два» приказало долго жить. И хотя к плате по-прежнему можно подключить 6 накопителей, не считая флоппи-приводов, но 4 из них должны быть подключены через *SerialATA* интерфейс — именно столько разъемов SATA установлено на плате. Параллельный же ATA-порт остался только один (в Intel решили, что больше в современном компьютере не нужно, и поддержку второго порта в южный мост чипсета включать не стали), так что потребителю предлагается всерьез задуматься о перспективах отживающих свое IDE-накопителей.

Претерпела некоторые изменения и привычная система охлаждения: разъем для процессорного кулера стал четырехконтактным, в соответствии с новой Intel'овской спецификацией. Кроме упомянутого разъема на плате предусмотрено управление еще тремя вентиляторами — двумя системными и вентилятором блока питания. Причем для более качественного управления, кроме термодатчика на плате, предусмотрено подключение еще одного датчика, который можно установить внутри корпуса. В отличие от многих других производителей AOpen не пожалел установить радиатор и на южный мост чипсета. Справедливости ради нужно сказать, что он там совсем не лишний: на платах от Intel не закрытый радиатором чип раскаляется при работе так, что темнеет лак вокруг него. Ну, а то, что этот радиатор сделан из алюминиевого профиля, отображающего слово AOpen, — это уже, конечно, не более чем совпадение ☺...

Биокомпьютер — реальность и фантастика

Мини, микро... меньше некуда

Евгений ФИДЕЛИН

«В настоящее время, когда каждый новый шаг в совершенствовании полупроводниковых технологий дается со все большим трудом, ученые ищут альтернативные возможности развития вычислительных систем. Естественный интерес ряда исследовательских групп вызвали природные способы хранения и обработки информации в биологических системах. Итогом их изысканий будет гибрид информационных и молекулярных технологий — биокомпьютер».

(По сообщениям журнала PCWeek).

Транзистор (**рис. 1**) заменил радиолампу — и компьютер, прежде занимавший машинный зал, смог уместиться в тумбах письменного стола. На смену транзистору пришли интегральные схемы (**рис. 2**) — и вот мы носим компьютер в кармане...

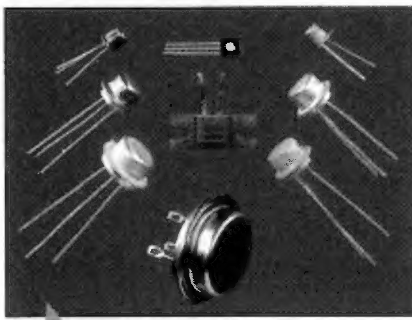


Рис.1

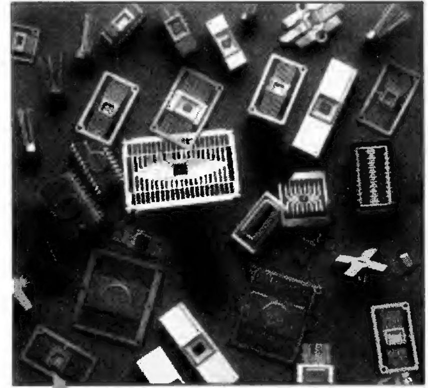


Рис.2

Интегральная схема — это десятки, а то и сотни тысяч активных элементов: триодов, диодов, конденсаторов, размещенных методами современной технологии в полупроводниковом кристалле размером менее ногтя мизинца — так называемом *чипе*. Микропроцессор такой же величины может иметь десятки миллионов активных элементов. Причем, размещение их в целях экономии места ведется в несколько «этажей». Однако ничего в нашем мире не дается даром. За запах цветов заплачена химической энергией растения, за шум прибоя — энергией ветра и волн, за красоту окружающего мира — энергией солнечного света. Платим мы и за миниатюрность компьютеров. Их компактность имеет пределы, полагаемые нынешними технологиями. И вот почему.

Компактность хороша не только сама по себе, но еще и потому, что она повышает быстродействие электронных приборов. В большом устройстве нужно больше времени, чтобы электроны пробежали свой путь по проводникам, накопили заряд в конденсаторах. Но, уменьшая размеры электронных устройств, мы должны сохранять на определенном уровне их мощность. Иначе выходные сигналы окажутся настолько слабы, что не будут восприняты вообще. А сохранение мощности, в свою очередь, приводит к повышению плотности энергии, выделяемой в единице объема.

Мы здесь сталкиваемся примерно вот с какой ситуацией. Вы вышли погулять в ветреную погоду. Пока ветер не очень силен, вы легко преодолеваете его сопротивление. Но ветер все усиливается, превращается в ураган, и в конце концов прогулка становится невозможной — того и гляди вас унесет... Такая аналогия дает наглядное представление о процессах электромиграции. Суть этого процесса заключается в том, что при определенной плотности энергии, выделяемой в единице объема, сила тока в проводнике может достигать миллионов ампер в пересчете на квадратный сантиметр! Обладая такой силищей, электроны начинают «сдувать» атомы с их законных мест в кристаллической решетке. И дело кончается тем, что в каком-то месте проводник разрушается, перестает проводить электрический ток. Что делать?

Выход из положения нашел американский физик **Дж. Баркер**. Раз проводники становятся помехой на пути микроминиатюризации аппаратуры, то он предложил обойтись без них. Дж. Баркер предложил использовать вместо проводимости... туннельный эффект!

Суть дела тут вот в чем. Специалисты в области физики твердого тела обнаружили странное с нашей обыденной точки зрения явление: время от времени электроны появляются в тех местах полупроводникового кристалла, где их быть не должно. Их кинетическая энергия недостаточна, чтобы преодолеть потенциальный барьер запрещенных уровней — и все-таки электроны появляются. Каким образом?

Попробуем обратиться к помощи такой механической аналогии. Представьте себе шарик, который катится по ровному столу. На пути шарика — препятствие, барьер. Перескочить его шарик не может, для этого он должен обладать по крайней мере вдвое большим запасом энергии. Значит, шарик не может оказаться по другую сторону барьера? Может! Но только при одном условии: если на его пути мы проделаем в барьере отверстие, туннель.

В действительности, конечно, никто никаких туннелей не делает. Шарик-электрон оказывается по ту сторону препятствия — потенциального барьера — только потому, что он обладает двойственными свойствами: с одной стороны он — частица, с другой — волна. И то, что не под силу электрону-частице, вполне доступно электрону-волне.

Вот этот-то туннельный эффект и предложил использовать Дж. Баркер. Поскольку, благодаря туннельному эффекту, электроны могут «просачиваться» сквозь полупроводниковое вещество, то зачем нужны проводники? Нужно вычислить вероятность перехода электронов из одного места в другое и допустить, что в данном месте — с такой-то вероятностью — у нас расположен проводник.

Революционность такого вывода трудно переоценить. Баркер, по существу, перевернул все традиционные представления классической электроники. В самом деле, представьте, каково было бы водителям, если бы перед мостом на обочине шоссе вдруг появился плакат: «Внимание! Мост функционирует с вероятностью 0.5». То есть, говоря иными словами, то ли он есть, то ли его нет в данный момент, никто не знает (есть в квантовой теории такой «принцип неопределенности». — *Прим. ред.*). Но вообще-то, если поток автомобилей будет непрерывным, есть шанс, что каждый второй доберется до другого берега.

Конечно, в обыденной жизни такая ситуация никогда не устроит. Но физики довольно часто руководствуются странными законами необычного мира микрочастиц и, тем не менее, достигают хороших результатов. Так получилось и в этот раз. Дж. Баркеру удалось убедить скептиков. Причем сделал он это довольно простым способом: он построил модель электронного блока, руководствуясь своими принципами. И модель эта оказалась вполне работоспособной!

Живой компьютер

И на этом история новой технологии, по существу, только начинается. Дело в том, что к этой же проблеме предельной микроминиатюризации с другой стороны подошли биологи. Еще в 1974 году они установили, что многие органические материалы способны проводить электрический ток. Более того, когда проходит электрический сигнал определенной полярности, органическое вещество способно переходить в другое состояние. То есть, говоря проще, внутри каждого живого организма работают словно бы миллиарды триггеров — электронных приборов, каждому из которых свойственны два устойчивых состояния — 0 или 1.

Причем, электроника живой клетки (**рис. 3**) имеет молекулярный уровень. То есть, своеобразным триггером может быть всего лишь одна молекула... Но ведь и Дж. Баркер предлагает создавать электронные приборы на атомно-молекулярном уровне! Нельзя ли объединить одно с другим?.. Ученые уже давно ведут работы на неисследованной территории.

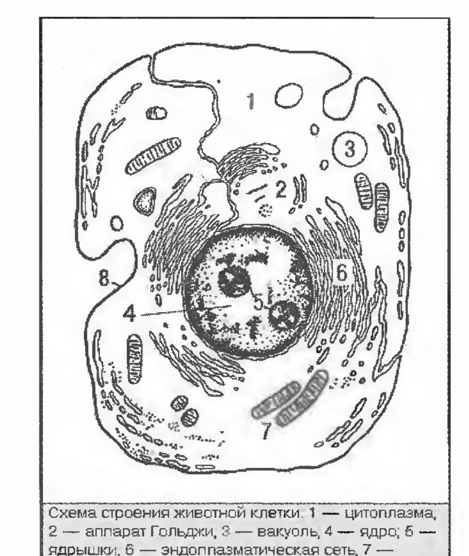


Схема строения животной клетки: 1 — цитоплазма, 2 — аппарат Гольджи, 3 — вакуоль, 4 — ядро; 5 — ядрышко; 6 — эндоплазматическая сеть, 7 — митохондрии; 8 — клеточная мембрана

Рис.3

Так, например, группа специалистов *лаборатории физики твердого тела в Орсе* (Франция), работающая под руководством **Д. Жерома**, интересуется сразу двумя видами материалов — проводящими полимерами и мономолекулярными кристаллами. Почему именно ими?

Мы привыкли к четкому разделению органического и неорганического миров. Трудно пока представить себе, что органические молекулы можно использовать для создания транзисторов. Однако ныне дела обстоят именно так. Более того, некоторые из органических веществ могут обладать даже свойствами сверхпроводимости!

Идея о сверхпроводимости в органических молекулах была высказана около полувека назад американским ученым **Б. Литтлом**. (*Сверхпроводимость* — физическое явление, наблюдаемое у некоторых веществ (сверхпроводников) при охлаждении их ниже определенной критической температуры Тк и заключающееся в обращении в нуль электрического сопротивления постоянному току. — *Прим. авт.*) Позже в лаборатории был получен класс органических веществ, способных хорошо проводить электрический ток при комнатной температуре. Основу этих проводящих молекулярных кристаллов составляли колонки из плоских органических молекул. После еще нескольких лет напряженной работы в лабораториях были получены и соединения, которые оставались проводящими при самых низких температурах, вплоть до 0.1 К (К — градус Кельвина, 0К = −273.15 °С. — *Прим. авт.*)

Таким образом была подготовлена почва для создания сверхпроводящих соединений, которые, как полагали в то время, могут существовать лишь при сверхнизких температурах.

Не меньшие достижения получены и в области органических проводящих полимеров. Тогда же американский физик **А. Хигер** продемонстрировал вентилятор, работавший от аккумулятора, пластины и рабочее вещество которого состояли только из органических веществ.

Какие же преимущества могут оказаться решающими при выборе между органикой и неорганикой? Их много. Органика, как вы уже поняли, открывает дальнейшие пути развития микроминиатюризации электронных устройств, вплоть до молекулярного уровня. Органика состоит из весьма распространенных в природе веществ. И наконец, органические молекулы обладают значительным разнообразием, способны к плавной перестройке, что может привести к созданию веществ с точным, заранее известным набором необходимых свойств.

Эти преимущества уже были «взяты на вооружение» природой в процессе возникновения живых организмов, и нет причин сомневаться в том, что они могут быть использованы и при синтезе систем с нужными нам сверхпроводящими или полупроводниковыми свойствами.

Отсюда, кстати, вытекает и еще одна особенность органической электроники. В одном из опытов подложку из кремниевого кристалла попытались заменить на протеиновую. Этот белок хорош не только тем, что вырабатывает электрические сигналы. Он является также родственным материалом многих клеток человеческого тела. А это значит, что протеиновые микрочипы можно будет имплантировать в человеческий организм!

Правда, как говорят скептики, у органической микроэлектроники проявляется очевидный недостаток: сложные органические молекулы, как правило, нестабильны, быстро распадаются. А значит, и основанные на них биологические микрокомпьютеры вряд ли будут очень долговечными. Однако мы знаем, как природа обошла эту трудность — она создала *самовоспроизводящиеся системы*. Именно к этому и стремятся специалисты в настоящее время.

В основе технологии современных полупроводниковых элементов лежит сочетание двух видов обработки этих материалов: введение так называемой *примеси n-типа*, способной отдавать электрон, и *примеси p-типа*, способной электрон принимать. Обычно такие примеси вводят в германиевый или кремниевый кристалл. Однако последние иссле-

дования в области органической химии показывают, как мы только что убедились, что это не единственные кандидаты для создания полупроводниковых элементов или чипов.

В присутствии паров йода полиацетилен тоже может получать свойства проводимости p-типа. Йод, проникая в структуру полимера, внедряется в нее и «ждет» появления электрона. (*Полимеры* (от греч. **poly** — «много» и meros — «доля», «часть») — вещества, молекулы которых (макромолекулы) состоят из большого числа повторяющихся звеньев. По происхождению полимеры делят на природные, или биополимеры, и синтетические. — *Прим. авт.*) То есть ведет себя точно так же, как, например, бор или галлий в кристаллической решетке кремния. А для того чтобы ввести в полиацетилен примесь n-типа, можно использовать литий, натрий или калий.

Органический полупроводниковый материал уже изготавливается в массовых количествах и стоит не дороже обычной клеенки. А значит, вполне серьезно можно подумать и об использовании его в электронике.

Первое возможное применение — создание *фотоэлементов*; для этого достаточно совместить один слой n-типа и два слоя p-типа. Следующее, возможно, самое перспективное сейчас применение — *OLED*, технология органических электролюминесцентных дисплеев (подробнее смотри в статье *Руслана Ризванова* «Органические мониторы», МК, №44(212) за 2002 г.) К настоящему времени удалось найти такие примеси, которые могут как отдавать, так и принимать электроны, оставаясь при этом достаточно стабильными. На их основе уже созданы электробатареи, которые показали не только достаточную долговечность, но и оказались почти в 10 раз легче обычных.

Созданы также биочипы, в которых используется способность молекул ряда органических веществ выпрямлять электрический ток. В частности, установлено, что молекулы *хинонов* могут иметь два электрических состояния, отличающиеся распределением электрических зарядов и зависящих от них водородных связей. Путем воздействия на хиноны электрическим током можно записывать логические нули и единицы.

Таким образом, уже сегодня начинают прорисовываться принципиальные возможности новой отрасли науки — биоэлектроники, которая, возможно, в этом веке придет на смену всем нам хорошо известной «обычной» электронике. Ведь уже сейчас есть биологические микроустройства, способные выполнять функции датчиков, процессоров, исполнительных механизмов. Родились новые разделы биофизики, которые изучают самоорганизацию биопроцессов. И если раньше исследователи шли в основном по пути воспроизведения в технике принципов, используемых в живых системах, то в настоящее время уже созданы гибридные системы и предпринимаются попытки, скажем, создать компьютер с процессорами, использующими некие элементы нервной системы. Его предполагают снабдить датчиками и исполнительными устройствами, в основе которых лежат молекулярные механизмы мышечного сокращения.

Интерес к биологическим микроустройствам в мире не случаен — источники биологических материалов дешевы и практически неиссякаемы, благодаря непрерывным методам культивирования микроорганизмов и животных клеток. Биоустройствами обеспечивается широкий спектр преобразуемых видов энергии — химической, механической, световой, электрической, а в ряде случаев возможна обратимость процессов преобразования — это позволяет использовать датчики для решения широкого круга задач. Биопреобразователи имеют сравнительно высокий коэффициент полезного действия (иногда почти 100%), благодаря успехам молекулярной биологии их можно конструировать с заранее заданными свойствами, избирательной реакцией и высокой чувствительностью. Эти и многие другие преимущества позволяют биологическим системам конкурировать с электронными, а по некоторым параметрам и превосходить их...

Фантастика становится реальностью

Основатель космической биологии **А. Чижевский** еще в тридцатые годы прошлого века сконструировал аппарат, который предупреждал о всплесках солнечной активности за несколько дней до начала очередной вспышки. Главной «деталью» этого прибора были крошечные бактерии, которые в зависимости от режима солнечного освещения меняли свою окраску.

Живые существа во многих случаях оказываются намного чувствительнее сложнейших электронных приборов. Так почему же тогда мы чаще всего пользуемся именно показаниями аппаратуры? Одна из главных причин — трудность дешифровки показаний живых «предсказателей», невозможность идентичной замены одного на другой.

Вот пример: сотрудники *биофака МГУ* как-то записали на осциллограф сигналы вкусовых щетинок комара-пискуна. При этом однозначно выяснилось, что каждому химическому соединению, которое комар пробует на «вкус», соответствует определенная электрическая последовательность импульсов, а сами датчики-щетинки срабатывали, даже если концентрация примесей составляла всего сотые доли грамма на литр воды. Эврика?! Не тут-то было... Стоило поменять комара, и характер электрических импульсов на экране тотчас менялся.

Другое дело, если мы переходим к опытам с микроскопическими живыми существами, а то и просто клетками. В опытах Чижевского замена одной пробирки другой вполне возможна — бактерий так много, что характерные черты отдельных особей нивелируются, сглаживаются, давая обобщенные результаты. Вот, оказывается, почему биофизики стремятся вести исследования именно на микронном уровне. Именно здесь появляется принципиальная возможность создания на биохимической основе отдельных элементов, а затем и целых систем — скажем, «биологических компьютеров» или «чувствующих роботов».

Конечно, такая проблема требует объединения усилий не только биофизиков и биохимиков, но и микробиологов, специалистов по генной инженерии. Совместными усилиями ученым удалось приспособить молекулы родопсина для создания биокомпьютера. (*Родопсин (зрительный пурпур)* — сложный светочувствительный белок, основной зрительный пигмент палочковых клеток сетчатки глаза у позвоночных животных и человека. Поглощая квант света, родопсин распадается и вызывает возбуждение зрительного нерва; в темноте вновь синтезируется. — *Прим. авт.*) Каким образом? Чтобы понять это, давайте на минуту обратимся к классическому компьютеру.

Любой вычислительный процесс состоит из трех этапов. Вначале возникает задача: исходные данные и представление, что с ними надо сделать. Затем следует собственно решение: вычисления в определенном порядке по строго заданным алгоритмам. И наконец, на третьем этапе полученное решение каким-то образом используется.

Но ведь нечто подобное сплошь и рядом происходит и в живой природе! По существу, вся жизнь состоит в решении задач и использовании полученных результатов. Проблемы нахождения пищи, ее потребления, усвоения, развития и размножения организмов — все это задачи, решаемые по определенному набору алгоритмов. А раз так, то можно, наверное, и компьютер сделать по тому же образу и подобию.

Именно такую цель и поставили перед собой японские специалисты, начавшие работы по специальному проекту, главная цель которого — создание «компьютера на основе элементов нервной системы, снабженного биологическими датчиками, а также исполнительными устройствами, использующими молекулярные механизмы мышечного сокращения».

Работы по созданию биопреобразователей уже миновали этап поиска — благодаря успехам молекулярной биофизики можно конструировать датчики с нужными свойствами, избирательностью и высокой чувствительностью.

Что же собой представляют датчики-преобразователи, которые являются в биокомпьютере источником входной информации? Мы как-то привыкли считать, что белок — нечто чрезвычайно неустойчивое, быстроразлагающееся... Однако его уже давно научились получать в кристаллическом виде, и тогда молекулу белка можно рассматривать как твердое тело с механическими свойствами, подобными, скажем, органическому стеклу или эбониту. Можно также белки и ферменты присоединять к специальным подложкам из бумаги, полистирола, нейлона, стекла или металла, после чего они опять-таки приобретают необходимую механическую прочность, с ними легко и удобно работать.

(Окончание следует)

Что у нас в карманцах, моя пр-релесть?

Алексей САЛО

Карманные компьютеры продолжают покорять сердца все большего количества людей, вынуждая их совершить покупку маленького электронного ассистента. А после приобретения наладонника у любого человека возникает множество вопросов, ответы на которые найти подчас не так уж легко. Надеемся, эта статья, выдержанная в стиле FAQ, облегчит вам поиск и поможет лучше разобраться со своим карманным помощником. Начнем с самого первого вопроса...

Где брать и чем просматривать книги?

Это, пожалуй, один из самых часто встречающихся вопросов, так как большинство людей покупает КПК именно для чтения книг. Благо книг и программ для их просмотра существует большое количество. Самой известной и распространенной программой для чтения является *HaaliReader* (**http://haali.cs.msu.ru/pocketpc/download.html.ru**). Это бесплатная



программа отечественного производства ☺, которая поддерживает формат *.fb2.* Литературу в данном формате можно взять по адресу **http://lib.coolparty.ru.** Для чтения книги другого формата (к примеру, *.doc* или *.html*) ее необходимо конвертировать в *.fb2* формат с помощью маленькой программы *AnyToFb2,* располагающейся по адресу **http://www.gribuser.ru/xml/fictionbook/2.0/software/Any2FB2.exe.** Кроме HaaliReader'а существует читалка от Microsoft под названием *MS Reader* (поддерживает формат *.lit*), **http://www.microsoft.com/reader/downloads/default.asp.** Книги в формате *.lit* можно взять с этой электронной библиотеки: **http://www.inlib.ru.** Для изготовления литературы данного формата понадобится небольшая утилитка под названием *Overdrive ReaderWork* (**http://www.readrworks.com**). И последней программой, которую мы рассмотрим в этой части, будет *RepliGo* (**http://www.cerience.com**). Уникальность ее заключается в поддержке очень распространенного сегодня формата *.pdf (Acrobat Reader).* Но на этом возможности RepliGo не заканчиваются. Программа также великолепно справляется с чертежами и электронными схемами.

Как распознавать русский рукописный ввод на Pocket PC?

Первая проблема, с которой сталкивается владелец новенького PPC, это отсутствие средств для ввода информации на русском языке (конечно, на рынке присутствуют и русифицированные КПК, но речь сейчас не о них). На сегодняшний день программ для решения этой проблемы предостаточно. Если со вводом текста на русском языке с помощью экранной клавиатуры проблем нет никаких вообще (возьмем, к примеру, программу *QKeys,* **http://www.pocketpc2002.ru/Products/downloads/QKeys/4/QKeysSetup40.exe**), то вот с рукописным вводом не все



так прекрасно. Дело в том, что процесс распознавания написанного от руки текста — это довольно сложная математическая задача, решить которую возьмется не каждая компания. По этой причине таких программ очень мало. Самые известные из них — это программа *PenReader* (**http://pocketam.narod.ru/pocketsoft/rasp/PenReader204.exe**) и *RussChar* (**http://www.hpc.ru/MacC/msd/russchar/download/RussChar.zip**). Если за первую программу придется отдать определенную сумму денег, то вторая программа распространяется как freeware, т.е. бесплатно. Обе эти программы работают по принципу встроенного в Windows Mobile 2003 *Letter Recognizer*'а, т.е. нельзя распознать сразу слово, возможно только побуквенное распознавание.

После установки программы ее необходимо «приучить» к собственному почерку (занимает эта процедура минут 10–20). Будем надеяться, что в скором времени появятся программы, распознающие русский текст так же хорошо, как и *Transcriber*, входящий в состав программ ОС Windows Mobile 2003.

Как и чем смотреть видео на КПК?

Рано или поздно этот вопрос встает перед каждым пользователем карманного компьютера. Одним просто интересно посмотреть, как вообще КПК справляется с видео, другие же покупают карманник, чтобы использовать его в качестве карманного домашнего ки-



нотеатра ☺. В любом случае, вопрос задан — а значит, нужно на него отвечать. В операционной системе КПК уже есть встроенные средства для просмотра видео (*Windows Media Player*), но в реальных условиях их не хватает. Нужно иметь более универсальный плейер, поддерживающий большое количество видеоформатов. Речь идет о программе *Pocket MVP* (**http://www.hpc.ru/soft/data/4167/PocketMVP_PocketPC.zip**), которая превосходно справляется с такими форматами, как *AVI, DivX, MPEG4, MP3* и *OGM.* Но одного плейера недостаточно для просмотра фильма, необходимо подготовить и сам фильм. Во-первых, нужно изменить разрешение фильма, чтобы смотреть его на экране размером 320×240 точек. Во-вторых, нужно немного сжать изображение, перекодировать звук и пр. Для этой цели служит одна очень удобная программа под названием *PocketDivXEncoder* (**http://divx.ppccool.com**). Чтобы перекодировать фильм для КПК с ее помощью, достаточно просто выбрать модель вашего КПК, и программа подберет оптимальные параметры кодирования.

Что такое эмуляторы, и где их взять?

Понятие «эмулятор» довольно обширное. В нашем случае это — специальная программа, которая «превращает» КПК в другое устройство, например, в игровую приставку *Sony PlayStation.* Или в компьютер с операционной системой MS-DOS. На сегодняшний день эмуляторов существует довольно много, и обычно это эмуляторы различных игровых консолей. Тем, кому посчастливи-

лось своими глазами созерцать такие легендарные платформы, как *Commodore64* и *Atori ST*, будет приятно увидеть их снова на своем КПК, загрузив из Интернета эмуляторы *PocketC64* (**http://www.hpc.ru/soft/data/5788/PocketC64setup.exe**) и *CastCE* (**http://perso.wanadoo.fr/castce/download.html**) соответственно. Для более молодого поколения существуют эмуляторы всем известной игровой приставки Денди. Самым быстрым из них является *PocketNester 0.7a*, который можно скачать по адресу **ftp://ftp.ware.ru/wince/801pn07_ppc2002.zip**. Если же вы желаете превротить свой карманный компьютер в современную игровую консоль Sony PlayStation, вам поможет в этом эмулятор *FPSEce*, **http://www.hpc.ru/soft/data/6408/FpseCE.zip**. Программа *PocketGnuboy*, располагающаяся по адресу **http://www.hpc.ru/soft/data/6711/pgnuboy118e.zip**, эмулирует приставку *GameBoy*. Есть еще два эмулятора, заслуживающие отдельного внимания: программа *PocketDOS* (**http://www.hpc.ru/soft/data/4485/PocketDOS_PPC_107.exe**), которая эмулирует операционную систему *MS DOS 6.22*, позволяя запускать все программы, написанные для ОС DOS, прямо на карманном компьютере, и программа *Windows XP v.2.60* (**http://www.hpc.ru/soft/data/4341/VindowsXP.zip**), которая превращает операционную систему Windows Mobile 2003 в привычную всем нам Windows XP.

Можно ли разгонять процессор, и чем?



Да, можно. К примеру, если Pocket PC КПК имеет процессор с частотой 200 МГц, то с помощью специальных утилит его можно розогнать как минимум в полтора роза (до 300 МГц). Еще утилиты подобного рода обладают очень полезной функцией замедления частоты. При замедлении частоты процессора автономное время работы КПК увеличивается (в большинстве случаев это даже важнее быстродействия). А на «замедленном» КПК можно спокойно читать книги, пользоваться органайзером и другими программами. Лучшей программой для разгона/торможения является *XCPUScalar 2004 2.6* (**http://www.hpc.ru/soft/data/5082/xcpuscalar.zip**). Кроме ручной настройки частоты здесь присутствует и автоматическая (в зависимости от потребностей программ). XCPUScalar интегрируется с такими оболочками, как *Wisbar*, *Icbar* и *Gigabar*, для мониторинга текущей частоты.

Какие есть программы для тестирования КПК?

В Интернете есть множество программ для тестирования карманного компьютера. Рассмотрим некоторые из них. Программа *PocketPC Mark* (**http://www.benchmarkhq.ru/cgi-bin/mobile.cgi?15**) позволяет протестировать КПК по следующим параметрам:

✓ *CPU Dhrystone* (производительность CPU Integer)

✓ *CPU Whetstone* (производительность CPU Float)

✓ *Memory Speed* (производительность Memory bus bandwith)

✓ *File System Benchmark* (тест производительности RAM File System)

✓ *Storage Cards Benchmark* (тест производительности карт iPAQ File Store)

✓ *Graphics* (тест графических библиотек Windows CE).

Интересной особенностью этой программы является то, что после получения результата его можно сравнить с производительностью других моделей КПК. Если кроме работы аппаратной части КПК вас интересует еще и работа компьютера в реальных условиях (например, скорость роботы КПК в таких приложениях, как *PocketWord* или *InternetExplorer*), тогда не лишним будет скачать мощный бенчмарк *Spb Benchmark* (**http://www.spbsoftwarehouse.com/products/benchmark/download.html?ru**).

Где достать бесплатное ПО для КПК?

Здесь мы не будем приводить список всех доступных сайтов, а ограничимся только некоторыми из них, самыми популярными. Описывать каждый сойт отдельно также нет смысла по причине их схожести: на всех ресурсах можно скачать программу на любой вкус. Из русскоязычных каталогов программ стоит обратить внимание на **http://www.winceware.ru** и **http://www.hpc.ru/soft**. Тем, кто владеет английским языком, следует посетить еще и такие сайты: **http://www.freewareppc.com**, **http://www.ipaqsoft.net/php/freeware.php**, **http://www.pdagold.com/software**. Даже если нужной программы во всех этих каталогах не найдется, нет повода для отчаяния — поисковые системы в Интернете еще никто не отменял.

Можно ли редактировать изображения на КПК, и чем?

Программ для редактирования/создания рисунков в Интернете достаточно, но не следует от них ожидать функциональности *PhotoShop*'а. Для рисования на КПК мы советуем воспользоваться программой *Pencil Box Delux* (**http://www.hpc.ru/soft/data/7137/PencilBoxDeluxe.exe**) компании *Mental Motion*. Эта программа имеет очень простой интерфейс, поддерживается 8 различных карандашей. Выполненный с помощью них рисунок очень похож на живой (сделонный реальными карандашоми), есть 6 страниц для рисования с быстрым переключением, имеется настраиваемоя сила нажатия и др. Для просмотра и редоктирования фотографий и других графических файлов но КПК трудно найти программу функциональнее *Spb Imageer* (**http://www.spbsoftwarehouse.com/downloads/imageer/SpbImageer1_setup_ru.exe**). Основные возможности программы:

✓ слайд-шоу в одно нажатие;

✓ просмотр миниатюр картинок в различных режимах;

✓ редактирование картинки;

✓ всплывающее окно в панели задач при вставлении карточки с картинкоми;

✓ оптимизация кортинок в пакетном режиме;

✓ публикация фотографий в Web;

✓ создание альбомов;

✓ заметки к фотографиям.

Любителям бесплатного ПО мы советуем загрузить с адреса **http://www.hpc.ru/soft/data/6277/UltraG.245.124.ARM.zip** программу *UltraG 2.4*. С ее помощью можно просматривать и редактировать изоброжения, а также наклодывать различные эффекты, изменять разрешение и пр.

Где достать новые темы оформления для Pocket PC?

В Интернете существуют целые сайты, полностью посвященные темам оформления экрана *Today* (аналог рабочего стола обычного ПК). Различия между этими сайтами заключается только в самих темах, структура же и наполнение примерно одиноковы. На каждом из нижеприведенных сойтов можно скачать кок обычные (статичные) темы, так и анимационные. Для пользовотелей, владеющих английским, будут интересны следующие сайты: **http://www.pocketpcthemes.com**, **http://www.microsoft.com/windowsmobile/resources/downloads/pocketpc/themes.mspx**, **http://www.pdagold.com/themes**, **http://www.pocketpcmall.com**, **http://www.pocketrocketfx.com/html/themes.htm**, **http://ppc4you.com/themes.cgi**. Всем остальным придется подучить английский, так как подобных сайтов в рунете нет (по крайней мере, мне не известны).

Хотелось бы узнать адреса лучших русскоязычных сайтов для пользователей КПК...

Вообще, сайтов по данной тематике в русскоязычном Интернете достаточно, что не может не радовать. Первым делом посетите сайт **http://www.hpc.ru**. Здесь можно почитать последние новости в мире КПК, статьи и обзоры, просмотреть просто огромный каталог карманных компьютеров с полным описонием каждой модели, загрузить разнообразные программы, пообщаться в форуме с более опытными пользователями — и доже совершить покупку КПК. Сайт обновляется ежедневно. Следующий веб-ресурс, **http://www.handy.ru**, предлагает вам посетить музей КПК, в котором можно посмотреть всю эволюцию карманных ПК, токже почитать разные статьи, пообщаться в форуме и, собственно, купить кармонник, если еще нет. Сравнить несколько КПК, прочитать результаты тестирования разных моделей, а также узнать мнения людей о том или ином карманном компьютере можно на сойте **http://www.pda1.ru**. Для людей, жаждущих общения или желающих решить возникшую со своим КПК проблему, создан большой форум, на котором каждый день общаются сотни, если не тысячи пользователей. Располагается этот проект по адресу **http://www.pocketz.ru**. Ну и в завершение приведем еще несколько адресов, посетить которые стоит каждому: **http://www.pdanews.ru**, **http://www.ihand.ru**, **http://www.pocketpcrussia.com**.

Базис и его настройка

Виталий ЯКУСЕВИЧ
santana@istc.kiev.ua
http://www.istc.kiev.ua/~santana

Продолжение, начало см. в МК, № 26–38, 40–43, 46, 50–52 (145–157, 159–162, 165, 169–171), 1 (172), 4 (175), 6–7 (177–178), 12–13 (183–184), 17–18 (188–189), 23 (194), 27 (198), 30 (201), 33 (204), 35 (206), 40 (211), 42 (213), 44 (215), 47 (218), 50 (221), 1–2 (224–225), 5 (228), 7 (230), 9 (232), 11 (234), 14 (237), 15 (238), 20 (243), 21 (244), 26 (249), 27 (250), 28 (251), 37 (260), 38 (261), 42 (267), 46 (269), 47 (270), 50 (273), 2 (277), 3 (278), 7(282), 10(285), 15(290), 21(296), 23(298), 27(302), 30 (305), 33 (308), 38 (313)

5. Локальные шины, арбитраж, режим Bus-Master и опция Peer Concurrency

Параллельная работо или, дословно, равноправная конкуренция. Так можно перевести ноименование данной опции. Этот параметр разрешает/запрещает одновременную работу нескольких устройств на PCI-шине. При активации опции включается дополнительное буферирование циклов чтения/записи в чипсете, которое ориентировано на PCI-шину. Также при этом состояние двух и более PCI-устройств сохраняется активным, а это означает, что запросы устройств выстроены в некую очередь и по возможности выполняются в соответствии с правилами арбитража в системе. Пусть и с дополнительным буферированием и ожиданием.

Полезность включения опции заключается и в том, что в некоторых случаях при передаче данных небольшими порциями PCI-шина получит к ним доступ без задержек, что маловероятно при обычной роботе нескольких устройств на шине. Но могут возникнуть проблемы, если не все PCI-устройства готовы поддерживать такой совместный режим работы. В этом случае работоспособность системы проверяется опытным путем.

Действие этой опции затрагивает и совместную работу PCI- и ISA-шин. Например, шинные PCI-циклы могут перераспределяться и буферизироваться во время ISA-операций, таких как передача по DMA-каналам в режиме BusMaster, что требует повышенного внимания со стороны системы.

Но необходимо отметить следующее. Описанный режим совместной и, что главное, равноправной работы устройств на PCI-шине не является постоянно действующим режимом работы системы. В зависимости от функционирования прикладного ПО, текущих условий работы устройств и требований системного арбитража работа PCI-устройств (и прежде всего в качестве master-устройств) может переводиться из стандортного режима работы в режим «равноправной» работы устройств (**PCI peer devices**). При этом действующие master-устройства теряют все свои полномочия. Лишается их и центральный процессор, кок один из возможных владельцев PCI-шины. Вследствие этого стандартно инициируемые процессором циклы CPU-to-PCI блокируются во время peer cycles, а сигнал BRDY# (**Bus Ready**) не выставляется. В это время осуществляются циклы роботы с основной памятью и кэшем и равнопровными PCI-устройствами.

Одним из возможных вариантов равноправного режима является способность процессора удерживать контроль над шиной во время PCI peer cycles с одновременным инициированием циклов работы с памятью.

Вот такой непростой режим, действие которого не так-то просто оценить с точки зрения необходимости использования. Параметр может принимать значения:

Enabled (по умолчанию) — розрешено;

Disabled — запрещено.

Опция может называться и *PCI Concurrency*, *PCI/ISA Concurrency* или *Bus Concurrency*.

Дополнительные устройства, охватываемые равноправной конкуренцией, появляются в опциях *PCI/IDE Concurrency* или *PCI-to-IDE Concurrency*. Понятно, что IDE-интерфейс — вовсе не дополнительная проблема. Конкуренция между ним и другими PCI-устройствами присутствует и в опции PCI Concurrency, просто в других опциях этот нюанс выделен особо.

Рассмотрим дополнительные интересные моменты, связанные с параллельной работой устройств, тем более, что с несколькими похожими опциями можно встретиться в одной системе. И остановим мы свое внимоние но чипсете SiS600. Приведем интересующую нос информацию из основных характеристик чипсета:

Provides High Performance PCI Arbiter

...

✓ Supports Rotating Priority Mechanism

✓ Supports Concurrency between CPU to Memory and PCI to PCI.

Concurrent execution between CPU, AGP and PCI Transactions

✓ CPU-to-DRAM read/write and PCI-to-PCI read/write

✓ CPU-to-DRAM read/write and PCI-to-DRAM read/write

✓ CPU-to-DRAM read/write and AGP-to-DRAM read/write

✓ CPU-to-DRAM read/write and AGP-to-PCI write (или PCI-to-AGP write)

✓ CPU-to-PCI write (or AGP-to-PCI write) and PCI-to-DRAM write (CPU-to-PCI, AGP-to-PCI write are temporary queued in multi-purpose FIFO if PCI-to-DRAM burst write is too long)

✓ CPU-to-AGP write (or PCI-to-AGP write) and AGP-to-DRAM write (CPU-to-AGP write, PCI-to-AGP write are temporary queued in multi-purpose FIFO if AGP-to-DRAM burst write is too long).

Достаточно внушительно и понятно, но почти такая же «внушительность» характерна и для некоторых других чипсетов, хотя далеко не для всех. Но вот такой предлагаемый спектр опций (*Memory Concurrency, PCI 33/66 Concurrency* и знокомая уже *PCI Concurrency*) встретился только в системах на чипсетах от SiS (SiS5600, SiS600, SiS620). В свое время чипсеты SiS530, SiS540 предложили подобные возможности, хотя и послабее, но среди двух опций была и опция PCI Peer Concurrency. Есть ли необходимость в таком количестве регулируемых параметров? Может быть, хватит одной обобщающей опции? Новерное, хватит. Но если возникают какие-либо проблемы в паре параллельных потоков, то тогда придется отключать весь механизм параллельности потоков в системе. Так что присутствие такого число опций в системе может оказаться вполне опровданным, тем более, что сложностей при работе с опциями нет. Они все предлагают значения **Disabled** и **Enabled**.

Сейчас остановимся на аппаратной поддержке приведенных опций и механизмов функционирования со стороны чипсета SiS600.

Регистр 50h Host Bus Interface control I

Бит 3 — CPU-to-Memory and PCI-to-Memory Concurrency Enoble

В состоянии «включено» CPU-to-Memory циклы и PCI-to-Memory циклы одновременно активны на системной шине и PCI-шине (путем буферирования). В отключенном состоянии параллельность токих циклов не осуществляется.

Бит 2 — CPU-to-PCI and PCI-to Masters Concurrently Access PCI Bus Function

Аналогично для циклов CPU-to-PCI и PCI-to-Memory. Информация из этого бита будет считываться, если бит 3 включен.

Бит 0 — PCI 33/PCI 66 Mosters Concurrently Access Memory Function

В данном случае речь идет о двух шинах, работающих на частотах 33 и 66 МГц, о master-устройствах на них и о параллельности запросов но доступ к основной памяти со стороны каждой из шин. Но интересный момент — при одновременности запросов от устройств на шинах более высокий приоритет остоется за PCI66 masters.

Регистр 51h Host Bus Interface control II

Бит 3 — Host to PCI33/66 concurrency Control

Регистр 80h Target Bridge to DRAM Characteristics

Окончание на стр. 26

Удачное знакомство

Олег КАСИЧ
kasich@mycomputer.ua

16 сентября в отеле «Санкт-Петербург» прошел IT-семинар, организованный компанией **BMS Trading**. Целью мероприятия было ознакомление участников с продукцией известных мировых производителей **Fujitsu Siemens Computers, Fujitsu, Intel** и **AOC International**.

Семинар начался с доклада Питера Хозера (**рис. 1**), директора по продажам в Восточной Европе Fujitsu Siemens Computers. Питер кратко рассказал о функционировании своей компании, ее структуре и организации производства. Более подробно он остановился на материнских платах Fujitsu Siemens. Их производство налажено на заводе в Аугсбурге (Германия), площадь которого составляет 180 тыс.м². Мощности завода позволяют изготавливать до 2.4 млн. плат в год. Размещение производства непосредственно в Европе дает покупателям определенные преимущества. В первую очередь это — ориентация производителя на качество продукции. Производственный брак материнских плат на заводе в Аугсбурге не превышает 0.4%, в то время как для Восточных компаний, занимающихся производством плат, этот показатель составляет более 1%. Такой результат достигается благодаря использованию высококачественных материалов и комплектующих, а также эффективной работе отдела R&D (Research & Development), состоящего из высококвалифицированных специалистов. К особенностям производства плат Fujitsu Siemens также можно отнести высокую степень автоматизации производственного процесса (более 95%) и ориентацию на «экологичность» продуктов (низкое содержание свинца и галогенов).



Рис.1

Выполнение таких требований при производстве продукции позволяет расширить сферу использования материнских плат данного производителя, которые применяются в настольных компьютерах, серверах, промышленных системах и медицинском оборудовании.

Расположение в одной временной зоне дает возможность получения оперативной технической поддержки, а развитая система логистики позволяет минимизировать сроки доставки продукции.

Помимо высокого качества, материнские платы Fujitsu Siemens отличаются рядом фирменных технологий. Повышенный интерес вызывает функция *Silent Fan*, которая позволяет контролировать обороты вентиляторов в зависимости от температурного режима в системе. Это дает возможность существенно снизить шум, издаваемый компьютером, если в данный момент он работает с небольшой нагрузкой (офисные приложения, Интернет и т.п.). При увеличении температуры процессора или внутри системного блока, обороты вентиляторов увеличиваются, соответственно улучшая охлаждение. Данная функция реализована при помощи дополнительного контроллера на плате, поэтому ее работа не зависит от установленной операционной системы. Питер Хозер продемонстрировал присутствующим работу этой технологии, используя систему с платой Fujitsu Siemens. Материнские платы оснащены также целым рядом других интересных функций, подробное описание которых требует отдельной статьи.

Перед присутствующими с докладом выступил Сергей Шевченко (**рис. 2**), специалист по применению продукции Intel. Освещая тему новой нумерации моделей процессоров, Сергей акцентировал внимание на том, что обозначения позволяют отличать друг от друга процессоры внутри некоторого семейства, но не имеют определенной математической зависимости и тем более не являются рейтингом производительности.



Рис.2

Значительная часть доклада была посвящена новой платформе *LGA775*, ее функциональности и преимуществам.

Сергей Шевченко также сообщил, что производство всей линейки процессоров *Pentium 4* в этом квартале будет переведено на ядро *Prescott* (ранее такой переход планировался на I-й квартал 2005 года). Очевидно, такие изменения связаны с успешным эксплуатированием 90-нм техпроцесса, т.е. высоким процентом выхода годных процессорных ядер.

Продолжил семинар Франц Капфенбергер (**рис. 3**), директор Fujitsu по продажам в Восточной Европе. Кратко остановившись на достижениях компании Fujitsu и ее подразделениях, Франц более подробно рассказал об истории развития жестких дисков и магнитооптических (МО) накопителей. Компания Fujitsu считает очень перспективным рынок 2.5" жестких дисков, поэтому не так давно были представлены первые диски такого формата с объемом 100 Гб (использовались две пластины). Также продолжают пользоваться большим успехом высоконадежные и производительные SCSI-диски, которые применяются в серверных решениях.



Рис.3

В настоящее время магнитооптика пользуется заслуженным вниманием. В отличие от обычных оптических носителей (CD, DVD), МО представляет собой значительно более надежный носитель информации. Объемы МО-дисков возросли до 2.3 Гб, и это далеко не предел. Также существенно повышено скорость передачи данных (8.5 Мб/с) и время доступа к информации (~20 мс). Учитывая невысокую стоимость хранения информации ($0.01 за 1 Мб) и удобство использования, можно признать, что МО-носители являются хорошим вариантом для создания резервных копий необходимых данных.

Завершал сессию докладов представитель компании AOC International Клаус Бургшталлер (**рис. 4**), заместитель директора по продажам в Европе. Компания AOC (Admiral Overseas Corporation) является подразделением *TPV Group*. Основана в 1967 году, как производитель цветных телевизоров на Тайване. В настоящий момент компания специализируется на производстве мониторов (CRT и LCD), LCD-телевизоров и плазменных дисплеев (PDP).

Группа TPV является одним из крупнейших производителей CRT- и LCD-мониторов, большая часть которых реализуется OEM-заказчиком, которыми являются известные мировые брэнды (в 2003 году только 12% продукции TPV было реализовано под брэндом AOC).

Рис.4

В настоящее время ассортимент AOC не очень велик, но вскоре его пополнят новые модели LCD-мониторов, которые будут анонсированы в начале 2005 года. Судя по слайдам, продемонстрированным Клаусом, в новых моделях огромное внимание уделено дизайну, который заслуживает похвал. После доклада Клаус Бургшталлер торжественно вручил дилерские сертификаты партнерам компании.

Далее последовала неофициальная часть мероприятия, где продолжилось общение с участниками семинара.



Софт-пробирка

Пингвин под окном

Эдуард ОЧЕРЕТНЫЙ

*Сегодня я расскажу вам о замечательном изделии японских программистов — **CooperativeLinux** (сокращенно **coLinux**). Девиз звучит примерно так: «Если Linux может совладать с любой архитектурой, то почему бы в качестве архитектуры не рассмотреть другую операционку?» Вот и взяли японцы обычное 2.4.*-ядро и собрали дистрибутив, который можно запустить из-под Windows. Все манипуляции я проводил в Windows XP Pro.*

Кооперативка

Итак, скачиваем с сайта разработчиков (**http://www.colinux.org**) установочный экзешник и устанавливаем colinux. Эта процедура затруднений вызвать не должна. После установки в систему инсталлируется сетевая карта, которая позарез нужна нашему Пингвину для полноценного взаимодействия с Виндой. Если возникли какие-то проблемы с созданием сетевушки, то следует произвести это вручную. Для этого кликаем правой кнопкой мыши по иконке *Мой компьютер*, выбираем пункт *Свойства > Оборудование > Установка оборудования > Да, устройство уже подключено > С диска* и указываем месторасположение файла **OemWin2k.inf** (по умолчанию лежит в каталоге **netdriver** корневого каталога программы).

Далее настоятельно рекомендую скачать образ **Debian** (**http://mesh.dl.sourceforge.net/sourceforge/colinux/Debian-3.0r0.ext3.1gb.bz2**, 17.5 Мб). Есть образы и других систем, но их размеры достаточно велики — от 80 Мб и больше. Стоит отметить, что при распаковке образ Debian займет примерно 1 Гб. Разработчики рекомендуют закинуть его в корневую папку программы. Не лишней будет заливка swap-имиджа, его образок объемом всего в 1 Кб можно загрузить с **http://gniarf.nerim.net/colinux/swap**.

Теперь осталось отконфигурировать файл **default.colinux.xml**. Обратите внимание на расширение данного файла — **.xml**. Весьма удобно. Откроем его в любом текстовом редакторе (я использую Word). Взглянем на структуру файла. Сначала описываются блочные устройства, а ниже общая конфигурация. Синтаксис описи устройств примерно таков: `<block_device index="X" path="\DosDevices\c:\imge_name" enabled="true" />` для виндовых дисков, и `<block_device index="X" path="\Device\HarddiskY\PartitionZ" enabled="true" />` для корня системы, которая уже установлена на диске. В предложенном примере **X** означает номер устройства начиная с нуля, **Y** — номер физического диска, могут принимать значения **0–3** (аналог *had-hdd*). Ну а **Z**, как вы уже догадались, — номер партиции, значения стартуют с **1**. Исходя из этого, *hda1* — это **\Device\Harddisk0\Partition1**, а *hdd4* — **\Device\Harddisk3\Partition4**. Тут же добавим CD-ROM, куда ж без него. Для этого следует ввести примерно такую строчку: `<block_device index="X" path="\Device\Cdrom0" enabled="true" />` (об **X** см. выше).

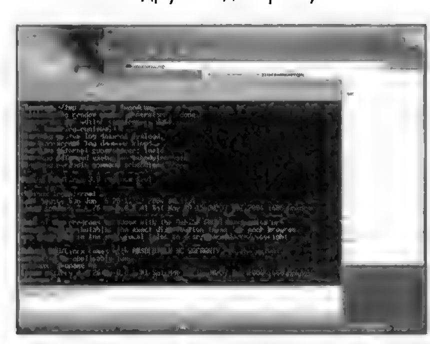


Дальше ничего особо редактировать не стоит, можно лишь увеличить размер оперативной памяти вашей Linux-машины.

Запускаем нашего Пингвиненка командой **colinux-daemon.exe -c default.colinux.xml**. Замечу, что это не единственное и далеко не самое идеальное обличье coLinux'а — его также можно запускать в NT-консоли (опция **-t nt**) или вовсе без терминала, с последующей возможностью подключиться по *ssh* или *telnet* (опция -d). По праву наилучшим ssh-клиентом является **PuTTY**. Скачать его можно здесь: **http://www.chiark.greenend.org.uk/~sgtatham/putty**.

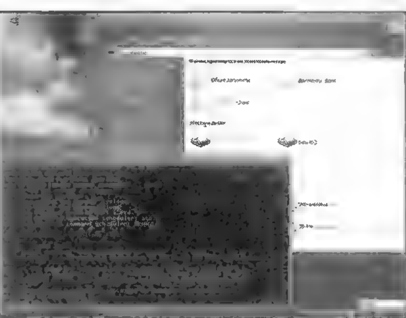
Топология без [illegible]

Всем хорош образ Дебиана, но софта в нем маловато. Несмотря на то, что он успешно отхватывает целый гектар, в нем отсутствуют такие весьма желательные пакеты, как *X-Window*, *KDE* и пр. Бороться с этим можно двумя способами: скачать все из Сети или просто использовать другой дистрибутив. Я по-



шел по второму пути, заказав в *Lafox.Net* дистрибутив **TopologiLinux 4.0.0**. Мой выбор не случаен, поскольку этот дистрик ставится прямо на виндовс-раздел в виде образа (а что нам еще надо). Сайт проекта можно найти по адресу **http://www.topologilinux.com**.

Установка Тополога проходит в два этапа: первый под Виндой, второй — непосредственно в Linux-среде. Для начала вставим первый диск в привод и создадим образ диска. Для этого следует нажать *Continue*, выбрать диск и задать размеры корневого и своп-файлов. В следующем окне следует отказаться от установки загрузчика (все равно не понадобится). Далее загружаемся с того же диска (разумеется, выставив в BIOS'е



опцию загрузки с CD-ROM), в ответ на приглашение загрузчика набираем **install**. Дальше все практически идентично установке *SlackWare Linux 9.1* (см. статью Сергея ЯРЕМЧУКА «Пингвин-патриарх», МК, 18–19 (293–294)), с тем исключением, что нужные разделы уже созданы. Еще маленький хинт, который упростит жизнь: сконфигурируйте coLinux'овскую сеть во время установки, а то потом замучаетесь лазить по файлам конфигурации. Это не смертельно, но зачем же ☺.

Первая загрузка проходит с CD, только на сей раз ничего набирать не надо, достаточно просто нажать *Enter*. Логинимся под рутом, создаем блочные устройства, которые будет использовать coLinux:

```
cd /dev
mknod cobd0 b 117 0
mknod cobd1 b 117 1
mknod cobd2 b 117 2
mknod cobd3 b 117 3
```

Эти команды создают файлы устройств, к которым будут монтироваться корневой раздел, своп, CD-ROM и флопик. Приводим файл **/etc/fstab** к следующему виду:

```
/dev/cobd0 / auto defaults 1 1
/dev/cobd2 /mnt/cdrom0 iso9660
users,defaults,noauto,ro,user 0 0
/dev/cobd3 /mnt/floppy auto
users,defaults,noauto,user 0 0
proc /proc proc defaults 0 0
/dev/cobd1 none swap sw 0 0
```

Перезагружаемся в родную Винду и правим **default.colinux.xml** так, чтоб корневым образом был файл **\tlinux4\base.img**. О том, как задать путь к другим устройствам, было написано выше. Запускаем coLinux и радуемся жизни. По идее, все должно работать, если нет, то стоит посмотреть соответствующие конфигурационные файлы.

Не консолью единой

Неполным будет мое повествование, если я не расскажу, как поднять иксы. Для начало немного теории. Во время старта системы *X-Window* запускаются несколько демонов, среди которых *X-сервер* и *X-клиент*. X-сервер общается непосредственно с оборудованием (такой себе драйвер). Так вот, если мы запустим сервер под Виндой, а клиент стартанет по требованию в coLinux'е то мы сможем любоваться красотами KDE в обычном окошке. Первым делом нам нужно установить одну замечательную штуковину под названием **CygWin**. Опять же у нас два пути: скачать все, что нужно, с сайта **http://www.cygwin.com**, или просто заказать в сетевом магазине. С установкой, я думаю, разберетесь. Полная установка занимает «гиг с хвостиком», но можно установить только пакеты, относящиеся к XFree86.

Непосредственная настройка производится путем редактирования пары файлов нашего Линукса. Прежде всего (на этапе загрузки) нужно запустить демон *XDM* — для этого добавим в конец стартового скрипта **/etc/rc.d/rc.S** строчку

```
exec /usr/X11R6/bin/xdm
```

Далее следует отконфигурировать XDM так, чтоб он мог принимать запросы по сети. Для этого нам потребуется снять комментарий со строчки `* any host can get a login window` в файле **/etc/X11/xdm/Xaccess** (следует снять оба символа # в строчке). Запускаем CygWin'овскую консоль и вводим такие строки:

```
X -query <coLinux IP> -once
```

Первый параметр задает адрес машины, второй — указывает на необходимость остановки сервера на удаленной машине после завершения сеанса связи.

Окончание. Начало на стр. 23

Управляет характеристиками 33 МГц PCI-моста, как целевого, при доступе к памяти.

Тем не менее, **Бит 1 — PCI Peer Concurrency.**

Включением обеспечивается параллельность потоков CPU-to-DRAM и PCI-to-PCI.

Стоит напомнить, что ведущую роль в обеспечении параллельности потоков в системе играют схемы арбитража, осуществляющие контроль за доступом устройств к памяти или PCI-шине с помощью Multi Transaction Timer чипсета.

Вот еще примеры некоторых опций: *Concurrent PCI/Host, CPU/PCI Concurrency, IDE Write Concurrency, CPU-to-DRAM Buffer Concurrency* (чипсет ITE8330Q). В последней опции речь идет о возможности совместного использования буфера CPU-to-DRAM и PCI-устройствами. Не самая лучшая аппаратная реализация. С расширением возможностей буферирования в системе применение подобных опций прекратилось. То же самое можно сказать и об опции *Concurrent Posted Write Buffers*. Как видим, вопросы параллельной работы устройств, интерфейсов можно рассмотривать значительно шире. К тому же эти вопросы и проблемы начали решаться достаточно давно, затрагивая не только системные контроллеры, но и периферийные мостовые схемы. Об этом нам напоминает давняя опция *ISA Concurrency*.

Аппаратную поддержку механизмов параллелизма и их поддержку через BIOS часто упоминаемая компания *Silicon Integrated Systems Corporation*, впрочем, как и другие компании, осуществляла и дальше. Поскольку раздельно программировались ровноправный доступ к памяти и к PCI-шине, то системы на чипсете SiS620 предложили такие варианты уже знакомых опций: *Concurrent Function [MEM]* и *Concurrent Function [PCI]*. Первая опция была посвящена равноправному доступу к памяти процессора и PCI master-устройств, вторая — параллельности циклов CPU-to-PCI и PCI-to-PCI. Значения опций: **Enabled**, **Disabled**.

Ликбез. Одной из характерных особенностей PCI-шины и ее системы мостов является возможность выполнения обмена данными между процессором и памятью одновременно с обменом между другими абонентами шины PCI — Concurrent PCI Transferring. Однако эта возможность реализуется не всеми чипсетами — а обычными абонентами шины (графические карты, контроллеры дисков и т.п.) используется редко.



Недреманное око

Марина и Сергей БОНДАРЕНКО
blackmore_s_night@yahoo.com

Чем является для вас Интернет? У каждого есть свой ответ но этот вопрос. Самое быстрое и свободное средство массовой информации, самое большое хранилище программ, самое большое брачное агентство... Да мало ли кок можно использовать Интернет! Он тоит в себе неограниченные возможности и, самое главное, его хватает на всех. Сегодня для многих он стоновится местом работы и средством для заработка денег.

В Интернете можно зароботывать по-разному, но практически всегда такая работа связана с посещением рознообразных сайтов. Например, если вы раскручиваете собственный проект, вам необходимо следить за его работоспособностью, а также постоянно публиковать свежие новости. Если вы являетесь профессионолом в какой-либо области, у вас наверняка есть в зопасе пара-тройка ресурсов, которые вы посещаете, чтобы быть в курсе последних событий.

Одним словом, так или иначе, каждый день приходится открывать одни и те же сайты и тратить время на их просмотр. И хорошо, если сайты обновляются регулярно. А если из десяти сайтов ноходится четыре-пять, на которых новостей нет, получается, что драгоценное рабочее (и, возможно, оплаченное) время потрачено впустую. Впрочем, если сайтов, которые вам необходимо посещать ежедневно, около десяти, с этим еще можно смириться. Однако если их несколько десятков, открытие и просмотр страничек на наличие обновлений становится задачей для самых терпеливых.

Программа, о которой мы хотим рассказать вам сегодня, позволяет экономить нервы, время и, соответственно, деньги (если у вас поминутная или помегабайтная оплата). Оно самостоятельно проверяет все занесенные в базу сайты и сообщает вам об обновлениях. Таким образом, вы избавляете себя от необходимости «холостых» загрузок. Название этой программы вполне соответствует ее предназначению — **Website-Watcher**. Этот «наблюдатель за сайтами» имеет такое огромное количество самых разнообразных настроек, что удовлетворит даже самого требовательного серфера. Программа может проверять сайты по расписанию, отсылать обновившиеся странички на указанный почтовый адрес, сохранять несколько версий одного и того же сайта и еще очень много всего. Но обо всем по порядку.

Робота с Website-Watcher начинается с выбора режима, в котором предстоит отслеживать изменения. В режиме, разработанном для начинающих пользователей *(Beginner)*, доступны только основные настройки программы, а в режиме эксперта *(Expert)* — абсолютно все. Переключаться между режимоми в дальнейшем можно через меню настроек *Options*. Советуем сразу же выбирать режим эксперта, так как только в нем можно по достоинству оценить всю мощь программы.

После удачного запуска можно начинать формирование базы ссылок, с которой программе предстоит работать. Сразу же заметим, что Website-Watcher может работать с несколькими базоми. То есть вы можете разделить ссылки на группы, которые нужно проверять одновременно. Например, у вас может быть одна база адресов, которые вом нужны по работе, а другая — сайтов, которые вы просматриваете для себя. Вы можете загружать эти базы поочередно. Также ноличие нескольких баз может оказаться полезным на рабочем компьютере, который не закреплен за одним пользователем.

Формировать базу ссылок можно двумя способами — вручную или же импортируя их из *Favorities* браузера. Первый способ предполагает ввод ссылки, а также ее описания в окно *Bookmark*, которое вызывается нажатием кнопки на панели инструментов программы или клавишей *Insert*.

Гораздо более удобный для многих способ формирования базы ссылок — импорт букмарков браузера. Website-Watcher поддерживает все основные популярные браузеры, в том числе *Internet Explorer*, *Netscape* и *Opera*. Для импорта ссылок необходимо нажать кнопку *Import or Export Bookmarks* на панели инструментов, после чего перетащить ссылки в основное окно программы при помощи мыши (**рис. 1**). Можно также перенести ссылки все сразу. При этом они будут сохранены в тех же папках, в которых они хранятся в Favorities браузера. Параллельно все ссылки будут помещены в папку *All Bookmarks*. Таким образом, вы сможете задавать проверку как всех ссылок сразу, так и только тех, которые находятся в текущей папке.

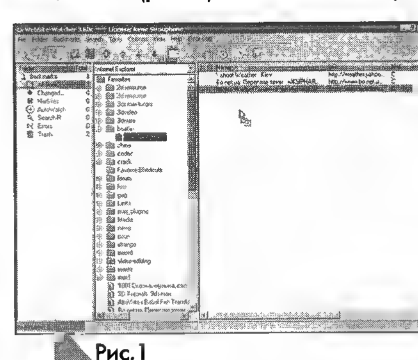


Рис.1

Так кок мы плавно перешли от создания базы к проверке ссылок, заметим, что при импорте букмарков Website-Watcher устанавливает для каждого Интернет-адреса параметры по умолчанию, которые подходят в большинстве случаев. Так, по умолчанию программа проверяет ссылки на наличие изменений содержания. То есть, содержание веб-странички сравнивается с закочанным ранее, и в случае несовпадения программа сообщает об этом пользователю. Есть и другие способы проверки — по дате и по размеру. В большинстве случаев они неэффективны для проверки веб-страничек и используются для обнаружения обновлений файлов на серверох.

Среди других параметров проверки можно отметить игнорирование чисел, HTML-тегов или определенных фраз. Например, на многих заглавных страничках сайтов отображается текущая дата и время. Понятно, что при каждой новой проверке это число будет изменяться, и программа будет сообщать вам об обновлении странички. Если же установить фильтр на изменение чисел, Website-Watcher будет пропускать эту информацию. Фильтры могут пригодиться также, если вы хотите узнать новости только по определенной теме. В этом случае вы можете ввести ключевые слова, на которые программа будет обращать внимание, а все другие фразы будут опускаться.

Как уже было сказано выше, проверять можно все ссылки сразу, по поппкам или даже по одной. При помощи привычных клавиш *Ctrl* или *Shift*, которые используются для выделения во многих приложениях Windows, можно вручную указывать, какие ссылки нужно проверять. Для того чтобы программа сообщила о наличии обновления на сайте, она должна загрузить страничку как минимум два раза. После первой проверки в строке статусо вы увидите нодпись *OK, initialized*, при дальнейшей работе об успешном выполнении операции Website-Watcher будет сообщать коротким *OK*. В строке статуса можно увидеть, какие ссылки уже были проверены, какие проверяются в данный момент и т.д. В главном окне программы доступна и другая информация — о времени последней проверки, о дате последнего изменения сайта и пр. Для того, чтобы проверить программу на работоспособность, советуем внести в базу ссылок какой-нибудь сайт, который обновляется постоянно (нопример, ленту новостей).

Не лишним будет скозоть о скорости проверки. Трудно себе даже представить, сколько времени нужно на загрузку в браузере восьмидесяти ссылок. Website-Watcher спровляется с базой такого объема минут за восемь-десять при работе на обычном диалапе 33.6 Кб/с.

Обратите внимание: если при роботе с Интернетом вы используете прокси, перед началом проверки не забудьте указать имя хоста и порт прокси в настройках Website-Watcher. Вручную это делать не обязательно — прогромма может использовать данные Internet Explorer. Для овтоматического экспорта настроек нужно нажать на кнопку *Use Settings from Internet Explorer*.

Если прогромма обнаружит на сайте изменения, он будет выделен жирным шрифтом. В этом случае вы сможете открыть строничку прямо в программе и просмотреть изменения. Это можно делать как по ходу проверки, так и после ее окончания (даже если вы находитесь в оффлойне). Если вы дождетесь концо проверки, вы увидите, что все сайты, но которых обнаружены изменения, вынесены в начало списка для ноглядности.

Website-Watcher хранит все загруженные им ссылки, поэтому вы можете просмотреть как новую версию сайта, так и старую. Но, конечно, самый удобный вариант просмотра — это с выделением изменений. Всю новую информацию на строничке программа выделяет желтым цветом, так что вам не придется долго искать обновления (**рис. 2**).

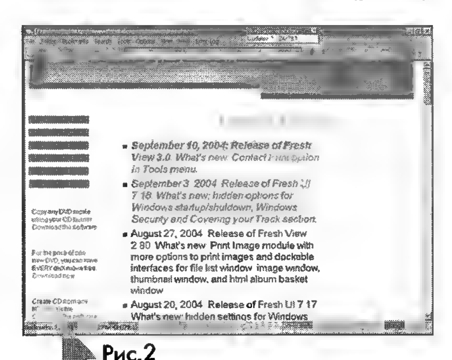


Рис.2

Панель для просмотро строничек напоминает обычный браузер. Вы можете просматривать сайты в полноэкронном режиме или же держать в поле зрения и загруженную страничку, и список проверяемых ссылок (**рис. 3**). Еще один вариант просмотро — старая и новая версии странички одновременно. В случае необходимости вы также можете открыть текущий сайт во внешнем браузере.

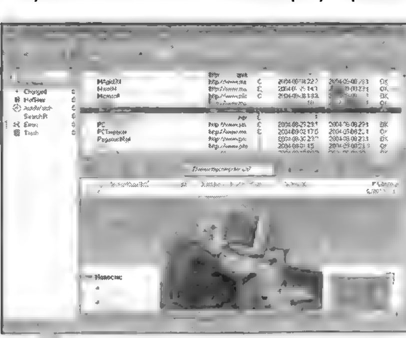


Рис.3

Если вы роботаете с сайтами, которые обновляются по нескольку раз в час, вам наверняка понровится функция автомотической проверки через зоданные промежутки времени. При этом можно указать также определенные дни недели и чосы, когда загружать страничку не нужно. Например, прогромма может проверять обновления на сайте автоматически каждый час с понедельника по пятницу с девяти утра до шести вечера. Удобно и то, что стронички с обновлениями можно получать по почте. Таким образом, вы можете следить за последними новостями, даже если на вашем компьютере нет доступа в Интернет, о есть только электронная почта. Для этого нужно запустить Website-Watcher на компьютере, который подключен к Интернету, укаэоть адрес, с которого будет отпровляться информация, SMTP-сервер, данные доступо к почте и адрес получателя. Вы можете получать короткое сообщение с адресом сайта, на котором произошли изменения, страницы с обновлениями или даже обе версии сайта — старую и новую. Параметры отправки почты нужно вначале зодать в общих свойствах программы, а затем дополнительно укозывать для каждой ссылки в ее настройках.

Website-Watcher можно использовать и как базу данных информации, которая может пропасть с сайта. Не секрет, что на многих веб-страничках найти новости, которые исчезли с титульной строницы сайта, бывает очень тяжело. Их могут переместить в зокрытый орхив или вовсе удолить. Если для вас вожны все новости, даже недельной или месячной давности, вы можете создоть собственный архив. По умолчанию Website-Watcher хранит только две последние версии проверенных им ресурсов, однако это число можно увеличить. В архиве можно хранить любое количество версий одного и того же сайта. Для занесения странички в орхив необходимо выбрать соответствующую команду в контекстном меню ссылки (**рис. 4**) или же воспользоваться сочетанием клавиш *Ctrl+F4*. Еще один способ — занести строничку в архив прямо из браузера. Для этого нужно открыть нужный сайт и щелкнуть по иконке в виде глаза, которая будет постоянно находиться в правом верхнем углу окна, когда Website-Watcher запущен.

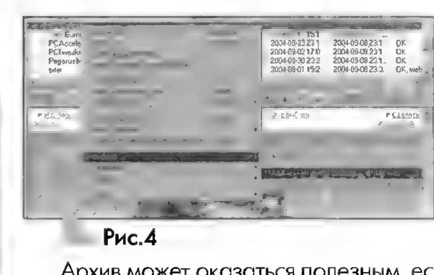


Рис.4

Архив может оказаться полезным, если понадобится найти какую-нибудь информацию. Даже если он разрастется до десятков мегобайт, вы легко сможете найти то, что нужно, при помощи удобной опции поиска. Кроме этого, базу данных можно экспортировать в фойл *CHM*. Это позволит работать с сохраненными строницами даже но компьютере, на котором не установлен Website-Watcher. Программа создаст список ссылок и добавит все файлы, которые к ним относятся. Учтите, что для экспорта вам необходимо будет предварительно скачать бесплатную программу *HMTL Help Workshop* с сайта Microsoft (**http://go.microsoft.com/fwlink/?LinkId=14188**).

Если же в вашем архиве наберется достаточно много информации, которая нужна не очень често, можно для экономии дискового пространства создать резервную копию и сохранить все странички в виде Zip-файла. Сбекапить можно не только находящиеся в архиве версии сайтов, но токже и бозу ссылок Website-Watcher, а также нострйки программы. Резервное копировоние также поможет восстоновить данные, если произойдет сбой в программе.

Что еще можно сказать об этой программе? Как и любой хороший, качественный продукт, который прекрасно справляется со своим заданием, очень скоро после первой установки Website-Watcher поселяется на панели быстрого запуска или даже в автозагрузке, и спустя некоторое время работу без него представить просто невозможно. Свидетельством того, насколько полезной нашли эту программу пользователи всего мира, является то, что интерфейс Website-Watcher уже переведен более чем но двадцоть языков, в том числе — но русский и укроинский. В последней версии программы эти дво языка даже включены в дистрибутив, то есть скачивать дополнительно языковые модули не нужно.

Скачать саму программу можно с сайта разработчика по адресу **http://aignes.com/download.htm**. Текущая на момент написания статьи версия — WebSite-Watcher 3.60d. Мы не даем прямую ссылку, так как уже есть бета-версия новой, четвертой версии прогроммы и, возможно, когда вы будете читать эту статью, будет доступен финальный релиз.

Цифры в образах

Надежда БАЛОВСЯК
nadia123@yandex.ru
http://www.nadia.ifyr.net

Известно, что большинство людей воспринимают информацию визуально, из образов окружающего мира. Психологи выделяют три типа людей — визуалы, аудиалы и кинестетики. Это личности, которые предрасположены к восприятию информации соответственно при помощи зрения, слуха и на ощупь. Но информация, представленная графически (в виде схем, графиков, диаграмм — в зависимости от типа данных), воспринимается лучше всего. Поэтому такой популярностью пользуются программы визуализации данных.

Разнообразных программ для визуализации очень много. Среди них можно встретить и маленькие утилитки, которые только и умеют, что строить график функции. Но сюда же относятся и такие программные монстры, как *Microsoft Visio* или *Origin*, возможности которых порой кажутся безграничными.

Попробуем вначале вывести общую классификацию программ для визуализации.

К первой группе отнесем программы, позволяющие «увидеть» числовые данные. Эти программы предназначены для визуализации (графического представления) данных в числовом (количественном) или аналитическом представлении. Такие программы могут графически отобразить числовые ряды или функции. Самое простое применение таких приложений — построение графиков функций и числовых рядов. Но, кроме этого, такие программы имеют ряд возможностей в области анализа и математической обработки графиков. Это — вычисление производной, площади криволинейной трапеции. Иногда присутствуют средства аппроксимации.

Вторая группа программ предназначена для отображения баз знаний и логических структур. Работа их состоит в построении различных организационных схем и диаграмм. Сюда относятся и программы для построения блок-схем, разных зависимостей и карт знаний (mind map).

Программы для построения временных шкал можно либо отнести к отдельному классу программ, либо представить их как разновидность продуктов предыдущего класса. Временные шкалы — это способ построения хронологической информации в виде графика: на оси X откладывается время, а на оси Y — категории событий. Сами события отображаются в виде отрезков графика. Сфера применения таких программ — это в первую очередь история, но сейчас временные шкалы стали использоваться и в менеджменте для отображения различных расписаний, наглядного представления параллельности и зависимости выполнения заданий проекта или при планировании дел.

В нашем обзоре представлены продукты, позволяющие увидеть все многообразие возможностей программ для визуализации. Мы исключили из рассмотрения только продукты для построения карт идей (mind map), поскольку разнообразие функциональных возможностей этих программ, ставших особо популярными в последнее время, настолько велико, что они заслуживают отдельного рассказа.

Графика для математиков

Начнем наш обзор с программ, рассчитанных на использование математиками, физиками, инженерами — то есть теми, кто работает с функциями и числовыми рядами.

Origin

Разработчик: Origin Lab
Сайт разработчика: www.originlab. com
Операционная система: Windows 9x–XP
Объем дистрибутива: 7.58 Мб
Адрес дистрибутива: http://www.originlab.com/index.aspx?s=12&lm=+62
Условия распространения: commercial product

Одним из многофункциональных пакетов, предназначенных для визуализации данных, является Origin. Программа умеет строить диаграммы различных типов для данных, представленных в виде таблицы. После запуска программы пользователь заполняет данными таблицу значений, после чего, выбрав тип представления данных, отображает их на листе Origin. В программе можно использовать различные типы представления данных — графики, диаграммы (линейчастые, круговые, кольцевые, гистограммы, поверхностные). При заполнении таблицы данных можно использовать формулы, получая таким образом таблицу значений функции, представленной в аналитическом виде.

Origin — единственная из программ нашего обзора, позволяющая не только строить график по заданному аналитическому или числовому представлению функции, а и выполнять обратное действие. В Origin можно построить график вручную (средство *«draw data»*), после чего отобразить таблицу значений для полученного графического представления.

Пользователям предоставлены широкие возможности форматирования графика — изменение цвета и типа линий, отображение легенды, добавление надписей и различных графических примитивов, настройка вида осей.

Настройка интервала отображения данных возможна также в виде одного из параметров осей. Здесь же, в окне настройки оси координат, можно изменить тип графика. По умолчанию при построении графика используется тип *«Linear»* — точки, полученные из таблицы значений, соединяются прямыми линиями.

Также можно задавать различные виды соединений точек из таблицы данных (линиями, точками, векторами).

Программа может импортировать данные в различных представлениях. В таблицу данных Origin можно добавить фрагменты текстового файла либо рабочего листа MS Excel.

Созданный график можно сохранить в виде шаблона, а потом использовать для построения других графиков. Проекты Origin (так называются созданные с помощью программы файлы) можно объединять в один.

Некоторые возможности программы делают ее незаменимой в инженерных расчетах и научных вычислениях. Например, средство *«screen reader»* — возможность, позволяющая отобразить на экране координаты любой точки системы координат, которая может принадлежать или не принадлежать графику. Средство *«data reader»* показывает координаты выбранной на графике точки.

Origin позволяет осуществить математическую обработку графика. Есть средства сглаживания, аппроксимация графика полиномом (многочленом) или прямой. Для некоторых графиков можно построить усредненную кривую.

У Origin есть стандартная библиотека математических функций. При необходимости пользователь может пополнять эту библиотеку своими функциями. Программа поддерживает технологию обмена данными *OLE*, что позволяет вставлять объекты Origin в другие программы (Word, Excel) и динамически обновлять их содержимое.

Microsoft Visio

Разработчик: Microsoft
Сайт разработчика: www.microsoft.com
Операционная система: Windows 9x–XP
Объем дистрибутива: 271 Мб
Адрес дистрибутива: http://www.microsoft.com/office/visio/prodinfo/trial.mspx
Условия распространения: commercial product

В этом обзоре мы не могли пропустить графического монстра от корпорации Microsoft. MS Visio позиционируется разработчиками в первую очередь как продукт для отображения логических и организационных схем и диаграмм, но в программе есть возможности работы с графиками аналитических функций и числовыми рядами. Visio предлагает пользователям среду для создания графических объектов практически любого типа. Библиотеки MS Visio представляют собой огромный набор готовых графических примитивов. Здесь можно найти шаблоны для создания графических представлений любой специфики — от схем трубопроводов, контрольно-измерительных приборов до диаграмм организаций, блоковых диаграмм, маркетинговых диаграмм, карт, веб-диаграмм, диаграмм баз данных и т.д.

Программа интегрирована с внешними устройствами и цифровыми камерами. MS Visio является частью пакета *MS Office*, поэтому интерфейс и принцип работы с программой аналогичны работе с Office. Поддерживается технология обмена данными с другими продуктами MS Office (**рис.1**).

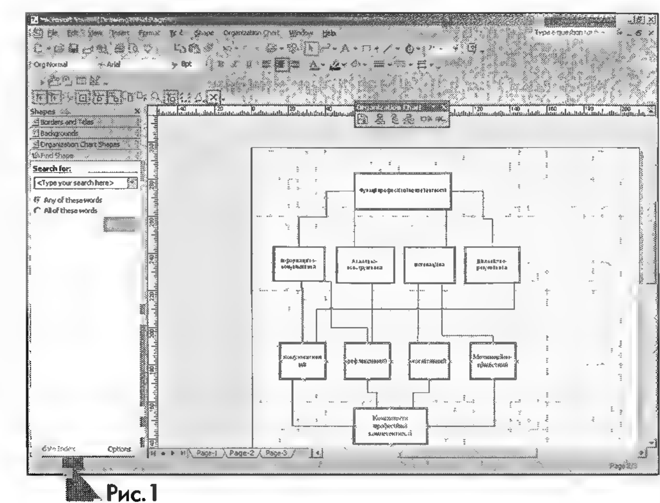


Рис.1

Программа владеет рядом возможностей, предназначенных для работы с графиками. Построив на рабочем листе MS Visio примитив графика, в специальном разделе *ShapeSheet* вы можете увидеть числовые и аналитические представления построенных графиков. Но в целом работа с графиками функций или графиками числовых рядов в MS Visio реализована неудобно. Можно только отредактировать уже существующие графические примитивы, добавляя к ним точки (так называемые вертексы), и для точек построить аналитические представления, то есть формулы, с использованием единиц измерения Visio. Пакет не обладает всеми необходимыми возможностями для работы с графиками аналитических функций, даже небольшие программы с этой задачей визуализации справляются гораздо лучше.

Advances Grapher

Разработчик: Alentum Software
Сайт разработчика: http://www.alentum.com/agrapher
Операционная система: Windows 9x/Me/NT/2000/XP
Объем дистрибутива: 1026 Кб
Адрес дистрибутива: http://www.alentum.net/files/agrapher.zip
Условия распространения: shareware

Кратко охарактеризовать программу можно так: универсальный построитель графиков с большими вычислительными возможностями.

В Advanced Grapher можно добавлять графики в нескольких представлениях. Программа может работать с графиками, построенными в обычной декартовой системе координат, причем есть возможность построения графиков, как обычных, так и обратных функций, то есть зависимости Y(x). Программа умеет строить графики в полярных координатах, графики с заданными параметрическими уравнениями, системами уравнений и неравенств, а также в табличной форме. Среди параметров графика, кроме его вида и параметров визуализации (стиль, толщина и цвет линий), можно задать параметры построения — интервал по каждой оси, количество шагов, максимальный разрыв. На графике можно отметить точки построения. При необходимости можно построить точечный график, то есть отобразить на графике только точки, без соединяющих их линий.

Если график невозможно представить в виде аналитической функции, Advanced Grapher построит его, базируясь на таблице данных. Если значения какого-то из столбцов таблицы являются результатами вычислений формулы, таблицу можно заполнить, указав эту формулу. При некорректном заполнении таблицы (например, если указана только одна из переменных), можно откорректировать таблицу значений. Мощные и удобные средства заполнения таблицы позволяют вставлять в нее пустые строки или удалять строки, менять местами значения X и Y, производить поиск и замену значений. А средство работы с буфером обмена позволят вставить в таблицу данные, сохраненные в формате MS Excel, либо содержимое таблиц MS Word.

Для просмотра промежуточных числовых значений отображенной на графике функции используется средство *«трассировка»*. С его помощью можно просмотреть любое из значений X и Y, перемещая горизонтальный бегунок.

В программе предусмотрены удобные средства масштабирования графиков, как каждого в отдельности, так и всех, построенных в одном документе.

Часть готового графика можно увеличить, указав интервал, который должен быть отображен на графике.

В программе сохранены некоторые наборы свойств по умолчанию, оптимальные для построения графиков. Существует как обычный набор — «свойство графика по умолчанию», так и «интервал по умолчанию» и «тригонометрический набор». Последний оптимален для построения тригонометрических функций, так как шкала по оси X в этом случае представляет собой не целые числа, а числа, кратные числу π.

AG не только предоставляет возможность графического отображения данных, но и является удобным средством для проведения математических вычислений. Для простых вычислений предназначен калькулятор, табло которого можно отобразить непосредственно в окне программы. Пользоваться калькулятором очень просто — достаточно ввести выражение и нажать *Enter*. Кстати, калькулятор «понимает» математические функции, которые используются в окне построения графиков. Синтаксис их описан в системе помощи программы (**рис.2**).

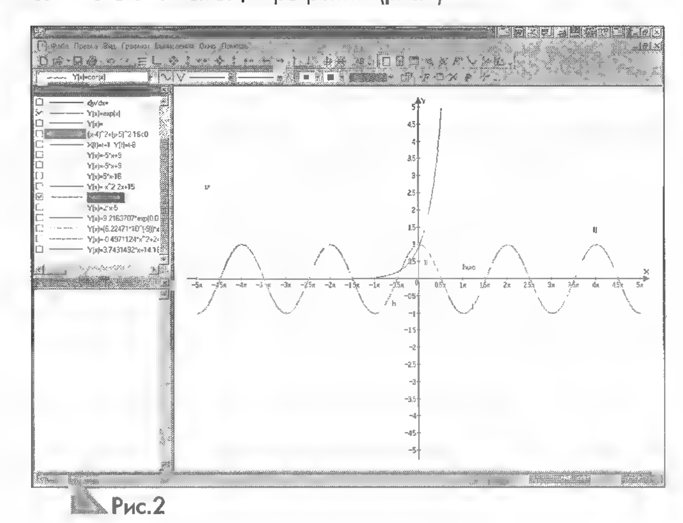


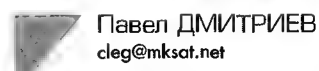
Рис.2

Но калькулятор — это только незначительная часть вычислительных возможностей Advanced Grapher. В полноценный математический пакет превращают программу другие важные функции. С помощью Advanced Grapher можно выполнять вычисления значений функции в определенной точке или подсчет таблицы значений, также можно исследовать функцию, определяя максимальное, минимальное ее значения и экстремум. Полученные результаты можно сохранить в файле или скопировать в буфер обмена. Среди других вычислительных возможностей программы — определение точек пересечения двух графиков, вычисление производной (с возможностью отображения ее на графике), построение касательной или нормали, вычисление определенного интеграла, регрессивный анализ. В программе предусмотрены средство импорта и экспорта данных. AG может импортировать таблицу из *.txt*- или *.csv*-файла и экспортировать таблицу в *.txt*-файл.

У программы многооконный интерфейс, в одном окне можно отобразить до 30 графиков.

(Продолжение следует)

ТеХничная верстка



Павел ДМИТРИЕВ
cleg@mksat.net

На прошлых занятиях мы уже неплохо научились работать как с отдельными символами, так и с целыми словами. Теперь пришла пора перейти к более крупным формам. У нас на очереди — абзацы.

Продолжение, начало см. в МК, №28, 30, 35 (303, 305, 310)

ТеХника красной строки

Сделать абзац просто — как мы уже говорили, для этого достаточно просто вставить в исходный текст пустую строку. Новый абзац как раз начнется с этого места. По умолчанию TeX верстает текст с выровниванием по ширине, используя переносы и изменяя ширину пробелов между словами так, чтобы правое поле было ровным. Если это удается, абзац считается хорошо сверстанным. При этом TeX никогда не превысит пределов сжатия, допускаемых текущим шрифтом.

Если же по какой-либо причине TeX постигла неудача в этом процессе, он выдаст предупреждение об ошибке. Оно будет выведено на экран и продублировано в **.log**-файл. В данном случае возможны две ошибки:

✓ слишком разреженная (жидкая) строка. Получаются такие строки из-за того, что предел «растяжения» строки оказывается все-таки превышен. TeX выдаст предупреждающее сообщение, начинающееся со слова *Underfull*;

✓ слишком длинная строка, которую не удалось достаточно сжать или сделать допустимый перенос. В этом случае строка частично «залезет» на поля. В этом случае предупреждающий текст будет начинаться с *Overfull*.

Есть один простой способ решить проблему длинных строк раз и навсегда. Он состоит в использовании специальной команды **\sloppy**. Она заставляет TeX предпочесть «жидкие» строки «переполненным». То есть после нее допускаются сколь угодно жидкие строки. Безусловно, TeX постарается, чтобы их было поменьше, но если придется выбирать между длинной строкой и жидкой, он выберет второе. В принципе, на некоторых текстах это может дать приемлемый результат. Иногда может получиться ужасно. Поэтому я бы не рекомендовал использовать эту команду в преамбуле документа. Достаточно использовать ее для одного проблемного абзаца. Для отмены этой команды можно либо использовать ее внутри группы, либо дать специальную команду **\fussy**. При этом учитывайте следующее правило: режим верстки абзаца определяется в тот момент, когда TeX считывает завершающую его пустую строку. Проще говоря, если вы решили использовать команду, влияющую на абзац внутри группы, то для того чтобы оно подействовала, необходимо, чтобы закрывающая группу фигурная скобка шла после пустой строки, завершающей данный абзац.

Второй способ борьбы с выравниванием состоит в использовании переносов слов. TeX делает это автоматически, но существует ряд моментов, в которых ему нужно помочь. По умолчанию в TeX включен режим переносов, принятый в Англии — переносится не менее 3 символов. Изменить это число на принятое у нас значение в 2 символа можно с помощью команды **\righthyphenmin=X**, где **X** — минимальное число символов, которые можно переносить. Кроме того, периодически (не слишком часто, но все же) встречаются слова, которые TeX просто не знает, как перенести. Кроме того, TeX не делает переносы в словах с диакритическими символами, словах с цифрами и т.п. Исправить это можно двумя способами:

✓ указав один раз, как нужно переносить конкретное слово, с помощью знаков **\-**: **ми\-ни\-ма\-лизм**;

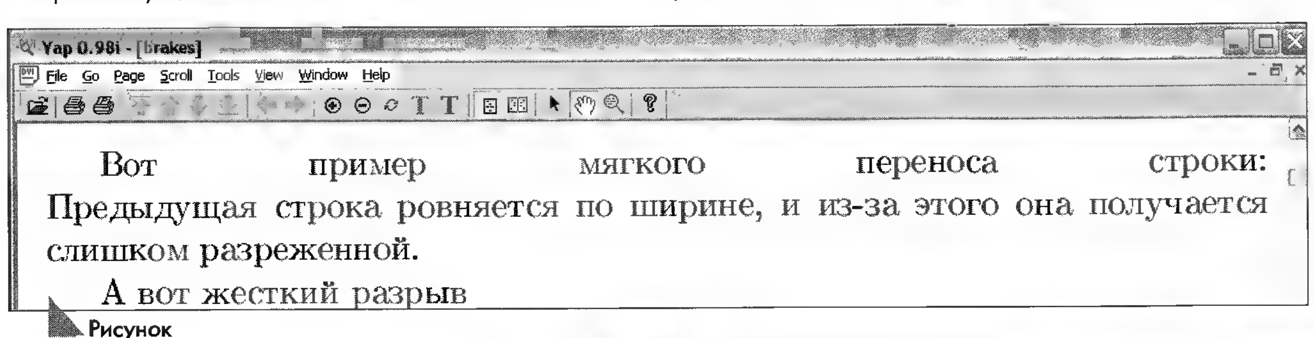
✓ добавив слово в «базу» переносов TeX с помощью следующей команды: **\hyphenation{ми-ни-ма-лизм неп-ра-вильно}**. В фигурных скобках после этой команды идет список слов, в которых с помощью дефиса указываются места переноса. Обратите внимание на тот факт, что при данных методах TeX будет использовать именно указанный вами вариант перевода, даже если он неправильный. При этом первый метод имеет больший «вес», чем второй.

Иногда можно победить «кривые» абзацы еще более грубым методом ☺. Для этого достаточно сделать перенос вручную. Это делается двумя способами. Первый — с помощью команды **\linebreak**, при этом получается «мягкий» разрыв, остаток строки до разрыва будет выровнен по ширине. Способ номер два состоит в использовании команды ****, которая делает «жесткий» перенос, не предпринимая никаких действий по выравниванию текста. В качестве необязательного аргумента этой команды можно указать TeX, сколько места оставить пустым в конце этой строки. Например, так: **\\[10pt]**. Также эта команда имеет вариант со звездочкой *****. Последний вариант даже более част в использовании. Помимо разрыва строки, он еще запрещает TeX заканчивать страницу на разорванной строке. Рассмотрим это на маленьком примере.

```
\documentclass{article}
\usepackage[russian]{babel}
\begin{document}
 Вот пример мягкого переноса строки: \linebreak
Предыдущая
 строка ровняется по ширине, и из-за этого она получается
 слишком разреженной.
 А вот жесткий разрыв\\ Текст остался неформатированным
\end{document}
```

Результаты представлены на **рисунке**.

В некоторых случаях может возникнуть необходимость обратного действия: вы можете захотеть оставить какую-то часть текста неразрывной. В принципе, мы уже рассмотрели основные средства, необходимые для этого, поэтому просто напомню. Первое — это неразрывный пробел, обозначаемый следующим образом: ~. Этот знак заменяется на стандартный пробел, но переноситься в этом месте строка не будет. Второй метод, позволяющий избежать лишних переносов, — использование команды **\mbox{текст}**, которая приказывает TeX считать весь текст, переданный в виде параметра, одной большой буквой. Соответственно, никакого переноса делаться не будет. Эту команду следует использовать с большой осторожностью, так как она при неумелом использовании может привести порой к очень неэстетичным результатам.



Рисунок

Ну, и еще один способ заставить TeX не делать переносы в отдельном слове был найден мною в книге *С. М. Львовского*. Он состоит в том, чтобы в конце слова, которое необходимо сохранить в целости, поставить команду **\-**. Этот способ, конечно, относится к приемам «жестокого» обмана TeX, но он дает приемлемый результат: TeX считает, что переносить это слово можно только после его окончания ☺.

Есть еще один «топорный» способ борьбы с некорректными строками. Этот способ состоит в использовании команды **\raggedright**. Она просто отключает выравнивание по ширине, оставляя правый край «рваным». Эта команда действует также на весь абзац, в котором была использована (см. замечание по использованию команды **\sloppy**). Также у этой команды есть побочный эффект: она отключает абзацный отступ. Если эту команду использовать внутри абзаца, вы не заметите ничего особенного, так как на этот момент абзацный отступ будет уже сформирован. Зато если использовать ее в преамбуле или внутри группы, результат будет налицо.

Вот, в принципе, основные (но далеко не все) приемы, позволяющие управлять выравниванием текста в TeX. Теперь перейдем к другим проблемам верстки абзацев.

Часто бывает так, что абзац не помещается на страницу из-за того, что он на 1–2 строки длиннее, чем нужно. В этом случае можно использовать команду **\looseness=число**. В данном случае число указывает, на сколько строк нужно «подогнать» абзац. Например, для того чтобы сократить абзац на одну строку, используйте команду **\looseness=-1**. Конечно, этот номер не пройдет, если в абзаце всего 2–3 строки, но при достаточно длинных абзацах TeX очень неплохо справится с задачей. Кстати, число может быть и положительным — абзац тогда будет увеличен на требуемое количество строк.

Прежде мы наблюдали, как работает TeX внутри абзаца. Теперь настала пора разобраться, как абзацы взаимодействуют между собой. Для этого сделаем небольшое отступление, чтобы поговорить о режимах, в которых работает TeX. Итак:

1. *Горизонтальный режим*. В нем TeX находится от первой встреченной буквы и до команды конца абзаца.

2. *Вертикальный режим*. В нем TeX находится при обработке команд, находящихся между абзацами и в преамбуле.

3. *Математический режим*. Используется при обработке формул, внутри соответствующего окружения.

Особенность работы в вертикальном режиме состоит в том, что в этом режиме все пустые строки и пробелы полностью игнорируются. Поэтому от конца предыдущего абзаца и до первой буквы следующего вы можете не беспокоиться о них ☺.

Кстати, существует еще один способ задания нового абзаца, помимо уже известного вам варианта с двумя пустыми строками. Это можно сделать при помощи команды **\par**, действие которой абсолютно аналогично. Следует отметить, что использование этой команды два раза подряд, а также после двух символов перевода строки, равно как и любая другая попытка добиться увеличения расстояния между абзацами с помощью дублирования команд задания нового абзаца, — абсолютно бесполезна. Команды задания нового абзаца не работают в вертикальном режиме. Для получения дополнительного интервала между абзацами следует использовать другие команды, приведенные ниже. Я так долго формулировал это правило, поскольку верстка межабзацных промежутков при помощи команд перевода строки — одна из основных ошибок людей, привыкших работать с WYSIWYG-системами типа того же Word'а.

Большинство отступов между абзацами TeX верстает автоматически, с учетом стилей и размеров шрифта, но в некоторых случаях возникает необходимость вмешаться в этот процесс. Три основные команды, используемые для этого, таковы:

✓ **\smallskip** — делает маленький отступ, примерно в половину строчной буквы высотой;

✓ **\medskip** — делает средний отступ, чуть больше, чем размер строчной буквы;

✓ **\bigskip** — формирует большой отступ, примерно с заглавную букву высотой.

Эти команды лучше всего использовать сразу после пустой строки, завершающей абзац.

Также вы можете задать размер вертикального промежутка в явном виде, используя команду **\vspace{число}**, параметр которой задается в TeXовских единицах размера и определяет величину отступа. Эта команда имеет также вариант со звездочкой, который позволяет создавать вертикальные отступы, даже если в этом месте проходит разрыв страницы.

Иногда бывает необходимо отменить абзацные отступы. Этот эффект легко достигается с помощью команды **\noindent**, которая должна идти прямо в начале нужного абзаца. Если эта команда располагается в середине абзаца или в его конце — она не работает.

Теперь пришла пора разобраться с таким понятием, как *интерлиньяж*. Это красивое слово обозначает просто расстояние между строками. По умолчанию TeX рассчитывает его автоматически, исходя из стилей и размеров шрифта. Но как это часто бывает, в этот процесс иногда приходится вмешиваться. Вообще-то менять межстрочные интервалы вручную крайне не рекомендуется, да и не нужно в большинстве случаев. Чаще возникает необходимость увеличить интервалы пропорционально. Допустим, вам понадобилось увеличить интерлиньяж на 25%. Для этого необходимо воспользоваться следующей конструкцией: **\renewcommand{\baselinestretch}{1.25}**. Для воздействия на весь текст поместите ее в преамбуле документа. Если зона действия — один абзац, то в его конце. В остальных случаях используйте *группы*.

Для управления интервалами между абзацами используется команда **\parskip=размер**. В данном случае интервал заданного размера будет добавляться к стандартному межстрочному интервалу. В большинстве случаев в изменении этого параметра нет никакой необходимости, так как TeX отлично справляется с этой задачей самостоятельно.

Довольно гибкий механизм верстки абзацев в TeX оказывается излишне громоздким, когда речь заходит о страницах. Если единичный абзац интерпретатор может легко загрузить в память и перебирать его варианты выравнивания, то попытка загрузить большой кусок текста и оптимизировать его разбиение на страницы чаще всего окончится неудачей: компьютеру просто не хватит на это ресурсов. Поэтому TeX в данном случае действует иначе. Как только он обнаружит, что текста хватит на страницу, он производит ее разрыв. Конечно, TeX может при верстке сместить разрыв страницы на пару строк в каждую сторону, но обычно этот процесс имеет «одноразовый» характер. Но у вас все же остаются в запасе несколько способов повлиять на верстку TeX'ом страниц. Во-первых, вы можете запретить TeX'у разрывать страницы. Делается это с помощью команды **\nopagebreak**. Если поместить ее в конце абзаца, то разрыв страницы в этом месте станет для TeX недопустимым кощунством. Кроме того, вы можете усложнить TeX'у процесс «разрывания» страниц. Делается это при помощи команды **\samepage**. Она разрешает TeX делать разрывы страниц только на границах абзацев. Переносы же внутри абзацев и между текстом и строчной формулой будут недопустимы.

TeX предоставляет и противоположную возможность: вы можете добиться принудительного разрыва страницы в нужном вам месте. Сделать это можно при помощи команды **\newpage**.

Для удобства верстки в TeX определено несколько интересных окружений, которые определяют стиль верстки заключенного в них текста.

✓ **quote**. Используется для вставки в текст цитат с соблюдением типографских правил. Текст в данном окружении смещается вправо, относительно поля, и верстается без абзацного отступа.

✓ **quotation**. Аналогично предыдущему, но используется для более длинных цитат. Текст верстается аналогично, но с абзацными отступами.

✓ **verse**. Служит для верстки стихотворных фрагментов и использует характерные переносы строк в строфе со смещением относительно левого края текста.

✓ **verbatim**. Верстается моноширинным шрифтом. Любые команды TeX, кроме команды завершения этого окружения, игнорируются.

(Продолжение следует)



Самострой

INSmалляцию строIM Sами

Дмитрий ПАВЛОВ
dpav@pisem.net

Читатель может спросить: «А мне-то зачем инсталлятор? Программ своих я не писал, и пока не собираюсь». В ответ на это расскажу такую историю из жизни. Был у меня компакт с довольно-таки старой игрой — Might and Magic VI, русская+английская версия. Да вот беда — русская версия оказалась совсем неиграбельна, глючила постоянно. А английская худо-бедно игралась. Пока я не вздумал патч поставить. Патч, как вы уже догадались, становиться упорно не хотел — не признавал некоторые файлы. Эти файлы пришлось скачивать из Интернета. В итоге патч все-таки встал, все было хорошо до поры до времени, пока из-за частого использования компакт не пришел в ужасающее состояние, и я захотел сделать копию. Вот теперь представьте: если бы я скопировал компакт 1:1, мне пришлось бы повторить всю последовательность операций: инсталляция русской версии — инсталляция английской версии — замена нужных файлов — установка патча. Не слишком ли громоздкая схема получается? Да наверняка и читателю знакома ситуация: программа вроде бы и работает, вот только файл надо туда-то скопировать, в реестр то-то записать... И так каждый раз при переустановке. Нельзя ли избавится от этой рутинной работы? Вот если бы как-то сохранить, а потом снова повторить уже готовый вариант...
Решением этой задачи я и хочу поделиться с читателями: использовать инсталлятор NSIS.

Итак, задумка проста — из уже «обработанной напильником» игры создать ее инсталляцию, учитывающую все проделанные изменения. При этом она должна:

✓ состоять из одного исполняемого файла;

✓ обладать удобным интерфейсом, при этом не слишком броским и перенасыщенным звуковыми и видеоэффектами (а вдруг во время инсталляции зайдет шеф/преподаватель? ☺);

✓ в начале работы выводить сведения о том, что за программный продукт инсталлируется;

✓ в диалогом режиме запрашивать директорию, в которую будет проводиться инсталляция;

✓ дать пользователю возможность выбрать лишь необходимые ему компоненты;

✓ каким-либо образом визуализировать собственно процесс инсталляции, который может быть достаточно долгим (чтобы пользователю не было слишком скучно), и обеспечить контроль над ним;

✓ не засорять самовольно реестр и системные папки Windows лишними записями и файлами (я уже и не говорю о программных модулях-шпионах);

✓ обеспечить создание корректного деинсталлятора.

Такие требования обусловлены спецификой моей задачи и личными пристрастиями, однако в целом они характерны для большинства инсталляторов. **NSIS (Nullsoft Scriptable Installation System)** идеально подходит. МК уже писал об этом великолепном продукте от не менее великолепной компании **Nullsoft** (см. в МК, №33 (256) «Генератор-инсталлятор» — Петр © «Roxton» СЕМИЛЕТОВ). С тех пор NSIS вырос до релиза версии **2.0**, взять его можно на официальном сайте разработчиков http://www.nullsoft.com/free/nsis. Среди новых возможностей — поддержка многоязычных инсталляторов, возможность представления компонентов инсталляции в виде дерева, система plug-in'ов для тех, кому не хватает стандартных возможностей. Внешний вид созданного инсталлятора теперь не ограничивается «фирменным» и легко узнаваемым стилем NSIS старых версий, а зависит лишь от вкусов его создателя. В частности, в комплект поставки включен набор файлов для реализации т.н. «современного» интерфейса, который весьма похож на привычные всем инсталляции от *InstallShield*. К уже существующим видам сжатия данных *(zip* и *bzip2)* добавлен *LZMA*, который, по утверждениям разработчиков, обеспечивает в среднем на 20% лучшее сжатие, чем bzip2. Изменились также и «внутренности» — добавлены десятки новых команд, изменены или вообще убраны за ненадобностью некоторые старые, появилась возможность использования пользовательских переменных в инсталляционных скриптах, переработана система управления страницами и т.д.

Прежде чем браться за работу, давайте попытаемся вспомнить, как проходит инсталляция (уверен, большинство читателей наблюдало этот процесс сотни раз). Обычно пользователю вначале предлагается ознакомиться и согласиться с лицензионным соглашением, потом — выбрать тип и необходимые компоненты установки, потом — папку установки, и, наконец, «откинуться на спинку кресла» и наблюдать за процессом копирования. Каждая из этих операций отображается последовательно в диалоговом окне инсталлятора, называемого в терминологии NSIS *страницей*. Не надо быть семи пядей во лбу, чтобы понять, что типовой инсталлятор состоит из четырех страниц: *показа лицензии, выбора опций инсталляции, выбора директории инсталляции* и *отображения прогресса инсталляции*. Поэтому при создании простейшего инсталлятора необходимо задать параметры для каждой из предопределенных страниц (впрочем, даже если пропустить определение некоторых из них, они будут заменены значениями по умолчанию, и инсталлятор все равно будет корректно работать), а также общие установки инсталлятора, затем скомпилировать — и результат перед вами! Если же какая-нибудь из предустановленных страниц не нужна — проблем тоже нет. Исключаем соответствующие установки, относящиеся к этой странице, — умный компилятор сам понимает, что вы не хотите ее видеть в готовом проекте, и просто-напросто выбрасывает страницу из конечного исполняемого файла. Если вы хотите соорудить инсталлятор с бо-ольшой претензией на оригинальность ☺, можете добавить свою, пользовательскую, страницу — это тоже несложно (хлопот, правда, будет чуть больше). В целом, создавать инсталляции с помощью NSIS необычайно легко — большая часть работы уже выполнена, и для 90% проектов костяк уже готов.

Со страницами, кажется, разобрались. Но как объяснить компилятору, что бывает несколько вариантов установки? Тут тоже все просто. Для обеспечения гибкости и управляемости процесса инсталляции в NSIS применяется *механизм секций*. Секция, в терминах NSIS, — это целостная, неделимая группа операций, производимых инсталлятором, таких как копирование файлов, создание и манипулирование ключами реестра и т.д. Секций может быть и несколько, причем они могут быть скомбинированы еще на этапе сборки, чтобы задавать предопределенные сценарии установки (например, *Минимальная, Обычная, Полная* и т.п.), или же пользователем непосредственно в процессе работы (вариант *Пользовательский — Custom*). Именно команды, определенные в секциях, выполняют всю основную невизуальную часть работы инсталлятора. Каждая секция имеет имя, определяющее текст, который пользователь сможет увидеть, выбирая компоненты для установки. Особое значение имеет имя секции *Uninstall* (или любое другое, начинающееся на *un*) — оно определяет, что данная секция содержит команды деинсталляции.

Итак, страницы и секции — это основа для создания инсталлятора, и этого обычно достаточно. Для гурманов же предусмотрен такой инструмент, как *функции*, которые и обеспечивают продвинутые возможности. Функции в NSIS бывают двух типов — *обычные (пользовательские)* и *функции обратного вызова (callback)*. Пользовательские функции хорошо знакомы всем, кто хоть раз пытался написать любую программу на любом языке программирования. Они играют ту же роль, что и при обычном программировании, нужны для выполнения серии одинаковых команд с разными параметрами и вызываются из самого скрипта. Функции обратного вызова (callback) также должны быть хорошо известны программистам под Windows. В отличие от обычных, их вызывает сам инсталлятор в ответ на некоторые события, происходящие в процессе установки. Например, функция `.onInit` вызывается вначале при инициализации инсталлера, `.onInstSuccess` — в конце процесса инсталляции при условии, что он был успешным, и т.п. Задача же программиста — предусмотреть реакцию на нужные ему события.

Из сказанного становится ясно, сколь многое можно сделать с помощью NSIS 2.0 — а ведь я еще и не говорил о командах! Как по мене, то предоставляемые ими возможности просто потрясают. Чтобы оценить их, достаточно пролистать более чем 100-страничный мануал, идущий в комплекте. Однако просто перечислять их скучно и никому не нужно — поэтому предлагаю самые важные из них опробовать в «полевых условиях», на рабочем примере — а я постараюсь прокомментировать все, чтобы было понятней.

Теперь оставим на время разбор NSIS и посмотрим на нашу проблему с другой стороны. Работоспособность программы в общем случае чаще всего зависит от двух факторов: наличия нужных файлов в определенных местах и наличия нужных ключей реестра в определенных его ветвях. Поэтому для любого инсталлятора жизненно необходимыми являются *списки файлов*, которые должны быть инсталлированы, и *ключей/записей реестра*, которые должны быть созданы. Для автора программного продукта никаких проблем нет — он и сам знает, что нужно для того, чтобы его собственное творение заработало. В других же случаях, в том числе и нашем, все не так просто. Иногда приходится проводить чуть ли не детективное расследование, чтобы заставить капризную программу работать.

Наш случай довольно простой — никаких других файлов, кроме как записанных в оригинальную инсталляционную папку, игре не требуется. В этом можно убедиться, просто скопировав эти файлы, например из архива, на компьютер, где игра не инсталлировалась, — она останется работоспособной. Иногда подопытная программа сама подсказывает: «Для работы мне нужно присутствие такого-то файла там-то и том-то». В тяжелых случаях поможет утилита **FileMon** — с ее помощью можно отследить все обращения программы к файлам и таким образом определить, чего же ей не хватает.

Обычно программы используют не только файлы, но и системный реестр Windows. Это следующее поле для наших исследований. Отсутствие нужных ключей в реестре для программы может быть таким же фатальным, как и отсутствие нужных файлов, а определить их сложнее. Но нам снова повезло — игра не требует обязательного присутствия «родных» записей в реестре (что подтверждается описанным выше экспериментом с «ручной» установкой игры на «чистый» компьютер). Однако это не значит, что она им вообще не пользуется (туда записываются различные параметры игры, и в этом скоро можно будет убедиться). Дело в том, что, не найдя эти записи при первом старте, игра заново создает несколько веток в реестре со значениями по умолчанию. Поэтому нам не нужно заботиться о реестре при инсталляции. В противном случае опять же на выручку приходит близнец уже упоминавшейся FileMon — утилита **RegMon** (обе программы и еще много других замечательных вещей на все случаи жизни можно взять на сайте http://www.sysinternals.com). Аналогично, она позволит отследить все обращения программы, но уже не к файлам, а к реестру.

Однако совсем избежать возни с реестром все равно нам не удастся. Программа-то дописывает данные в реестр автоматически, но сами собой они оттуда никуда не денутся. И если мы хотим создать корректный деинсталлятор, то надо предусмотреть удаление этих записей. Для решения этой задачи можно предложить два способа. Первый: взять *Regedit* и проверить в реестре наличие строк, которые, возможно, относятся к инсталлируемой программе. Способ второй: сделать то же, но с помощью специализированных утилит. Для этих целей идеально подходит **RegShot** (последняя версия — **1.7.2**; качать, например, здесь http://wasm.ru/tools/17/regshot.zip). Эта крохотная программа, будучи запущенной, позволяет в любой момент времени сделать моментальный «снимок» реестра, а также сравнить его с предыдущим, показав, таким образом, все изменения, произошедшие в реестре. Это нам и нужно. Запускаем RegShot, делаем снимок, запускаем игру, снова делаем снимок, сравниваем — и вот мы уже знаем, что программа пишет в реестр!

Итак, вооружившись нужными списками и знаниями о NSIS, можно соорудить полноценную, вполне работоспособную инсталляцию. Как я и обещал, после каждого логического блока команд я буду давать краткий комментарий, поясняющий работу соответствующей части скрипта.

```
; Simple MM6 Installation
; Version 1.2 – 03.08.2004
##### Defines:
 !define S_NAME "MM6"
 !define M_NAME "Might and Magic VI"
 !define F_NAME "Might and Magic VI: The Mandate of
Heaven"
 !define SOURCEPATH "E:\TEMP\MM6"
```

Здесь определены константы — сокращенное, среднее и полное имя инсталлируемого продукта (полезно, если надо будет по-быстрому, в походных условиях переделать этот скрипт в инсталляцию чего-нибудь другого) и путь, по которому уже можно найти игру (источник файлов). Кстати, пока будете экспериментировать со скриптом, полезно заменить реальные инсталлируемые файлы, которые занимают сотни мегабайт, на файлы-пустышки — файлы с теми же именами, но имеющие маленький размер. В противном случае каждый раз компиляция будет затягиваться на многие десятки минут.

```
##### Installer attributes:
 Name "${S_NAME}"
 Caption "${M_NAME} Setup"
 Icon "${S_NAME}.ico"
 UninstallIcon "${S_NAME}Un.ico"
 OutFile "${S_NAME}Setup.exe"
 SilentInstall normal
 BGGradient off
 InstallColors 0xFFC040 0x000000
```

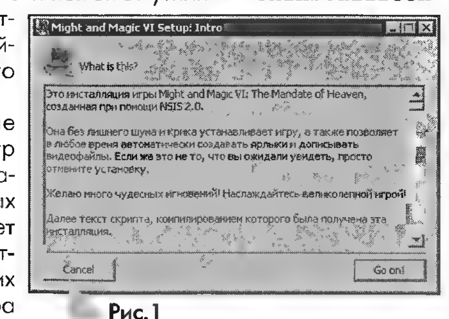


Рис.1

Определяются главные атрибуты инсталлятора: название, заголовок окна, конечный файл и файлы-иконки для инсталлятора и деинсталлятора (поскольку я не силен в рисовании, я просто скомбинировал иконку с головой дракона от самой игры со стандартными иконками Windows — компьютера и кор-

зины; потратил полтора часа, но, по-моему, вышло оригинально (**рис. 1**)). Тип установки определен как нормальный, с визуальным интерфейсом. Последние команды отключают вывод градиента и задают цвет фона и текста окна, где показывается процесс инсталляции. Я выбрал янтарный (0xFFC040) по черному (0x000000) — именно эти цвета у меня ассоциируются с игрой Might and Magic VI.

```
##### License page configuration commands:
LicenseText "$\rWhat is this?" "Go on!"
LicenseData "Readme.txt"
SubCaption 0 ": Intro"
```

Здесь собрано все, что относится к странице показа лицензионного соглашения. Поскольку в общем-то оно мне не нужно, то я приспособил эту страницу под информационные цели — показать, что именно инсталлируется, и заодно встроить сюда текст скрипта (тогда точно не потеряется ☺!) Поэтому я несколько изменил текст стандартной кнопки *I Agree* на *Go on!*, и окончание текста заголовка окна с *License Agreement* на *Intro*. Результат на **рисунке 1**.

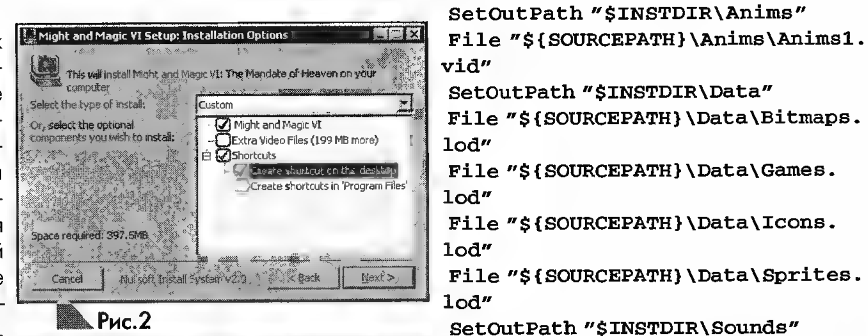


Рис.2

```
##### Component page
configuration commands
ComponentText "$\rThis will
install ${F_NAME} on your computer"
CheckBitmap "${S_NAME}Check.bmp"
InstType "Copy Required Files Only"
InstType "Create Shortcuts"
InstType "Full Install"
```

Продолжаем, на этот раз со страницей выбора компонентов. Задан текст в окне и картинка, определяющая вид выбранных/не выбранных/обязательных и т.п. компонентов. Опять же, не мудрствуя лукаво, я взял графику из набора поставки NSIS 2.0, заменив палитру на красно-оранжевые тона (я уже объяснял, почему). Смотрите, что получилось (**рис. 2**). Кроме того, я объявил здесь три стандартных типа инсталляции: копирование нужных файлов, то же — с созданием ярлыков, полная.

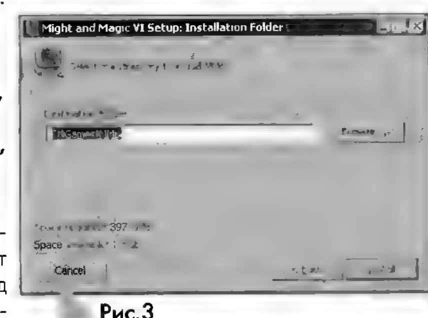


Рис.3

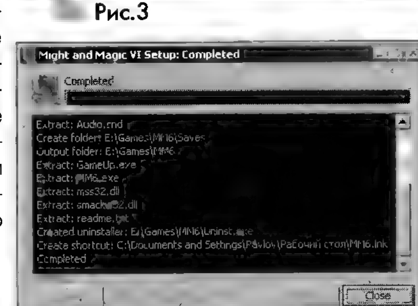


Рис.4




```
##### Directory selection page con-
figuration commands
DirText "$\rSelect the directory
to install ${S_NAME}"
InstallDir "E:\Games\MM6"
```

Страница выбора директории инсталляции, тут совсем просто. Опять определен текст в окне и путь инсталляции по умолчанию, который потом можно использовать через внутреннюю переменную $INSTDIR (**рис. 3**).

```
##### Install page configuration commands
AutoCloseWindow false
ShowInstDetails show
InstProgressFlags smooth
```

А это — установки для окна прогресса инсталляции. Определен детализированный показ процесса (люблю, чтобы все было под контролем ☺), плавная полоска индикатора прогресса и ожидание закрытия окна инсталлятора пользователем (**рис. 4**).

```
##### Installation execution commands
##### Change the options for the compiler:
SetCompress off
SetDatablockOptimize off
CRCCheck off
```

Эти команды определяют поведение компилятора и могут быть определены в любом месте скрипта, причем их действие начинается строкой ниже от того места, где они были определены, и продолжается до тех пор, пока не будет отменено другой командой. Здесь отменены все виды оптимизации (компрессия и оптимизация блоков данных), поскольку файлы игры MM6 уже сжаты, и дополнительные манипуляции ничего, кроме потери времени, не дадут. В другом случае я рекомендую отдельно проверить, дает ли включение этих опций выигрыш в размере для ваших файлов, и лишь потом делать выводы. Кроме этого я запретил самопроверку инсталлятором контрольной суммы (CRC) при старте (уповаю на надежность современных носителей ☺), поскольку она также забирает ощутимо много времени.

```
##### Section definitions
```

Дальше идет определение секций:

```
Section " ${M_NAME}"
SectionIn 1 2 3
 SetOutPath "$INSTDIR\Anims"
 File "${SOURCEPATH}\Anims\Anims1.vid"
 SetOutPath "$INSTDIR\Data"
 File "${SOURCEPATH}\Data\Bitmaps.lod"
 File "${SOURCEPATH}\Data\Games.lod"
 File "${SOURCEPATH}\Data\Icons.lod"
 File "${SOURCEPATH}\Data\Sprites.lod"
 SetOutPath "$INSTDIR\Sounds"
 File "${SOURCEPATH}\Sounds\Audio.snd"
 CreateDirectory "$INSTDIR\Saves"
 SetOutPath "$INSTDIR"
 File "${SOURCEPATH}\GameUp.exe"
 File "${SOURCEPATH}\MM6.exe"
 File "${SOURCEPATH}\mss32.dll"
 File "${SOURCEPATH}\smackw32.dll"
 File "${SOURCEPATH}\readme.txt"
 WriteUninstaller
"$INSTDIR\Uninst.exe"
SectionEnd
```

Итак, главная секция нашего инсталлятора. Очевидна ее структура: заголовок, набор команд, окончание. Команда SectionIn определяет, в каком из предустановленных типов инсталляции, обозначенных выше, будет принимать участие эта секция (в данном случае — во всех трех). Дальше идет серия команд копирования файлов: SetOutPath устанавливает путь, по которому будет произведено копирование файлов (используя шаблоны с $INSTDIR), File копирует необходимый файл (который включается в инсталлятор на этапе компиляции). Одинокая CreateDirectory создает директорию сейвов, а WriteUninstaller, как понятно из названия, производит запись деинсталлятора в основную папку с игрой.

```
Section "Extra Video Files (199 MB more)"
SectionIn 3
 SetOutPath "$INSTDIR\Anims"
 File "${SOURCEPATH}\Anims\Anims2. vid"
SectionEnd
```

Если вы все внимательно прочитали, здесь должно быть все понятно. Эта секция задействована только в третьем («Полном») варианте установки, и единственная ее задача — скопировать громоздкий видеофайл Anims2.vid в папку Anims.

```
Subsection "Shortcuts"
 Section "Create shortcut on the desktop"
 SectionIn 2 3
 CreateShortCut "$DESKTOP\${S_NAME}.lnk"
"$INSTDIR\MM6.exe"
 SectionEnd
 Section "Create shortcuts in Program Files"
 SectionIn 2 3
 CreateDirectory "$SMPROGRAMS\3DO"
 CreateDirectory "$SMPROGRAMS\3DO\Might and Magic VI"
 CreateShortCut "$SMPROGRAMS\3DO\Might and Magic
VI\${S_NAME}.lnk" "$INSTDIR\MM6.exe"
 CreateShortCut "$SMPROGRAMS\ 3DO\Might and Magic
VI\Uninstall ${S_NAME}.lnk" "$INSTDIR\UNINST.exe"
 SectionEnd
SubsectionEnd
```

Это часть интересна тем, что демонстрирует одну из новых возможностей NSIS 2.0 — создание древовидной системы выбора компонентов инсталляции. Формально, это две независимые секции, заключенные в оболочку из Subsection/SubsectionEnd. Именно это и определяет вид этих секций, потомков от общего родителя (**рис. 2**). По правде говоря, я бы назвал это не *Subsection*, а *Supersection*, так как по смыслу это не «подсекция», а «надсекция» — но разработчикам, наверное, виднее. В самих же секциях происходят тривиальные вещи — создание папок и ярлыков на *Рабочем столе* и в меню *Программы*, причем активно используются внутренние константы NSIS — $SMPROGRAMS (для доступа к меню «Программы») и $DESKTOP (для доступа к Рабочему столу). Вместе с еще одной, $ROGRAMFILES, они составляют наиболее широко используемые константы, вообще же их больше трех десятков.

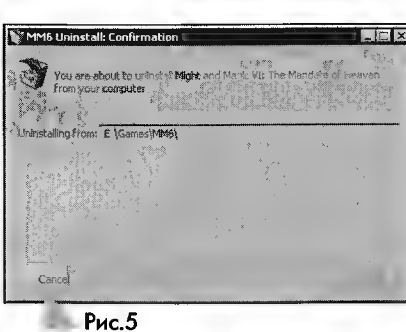


Рис.5

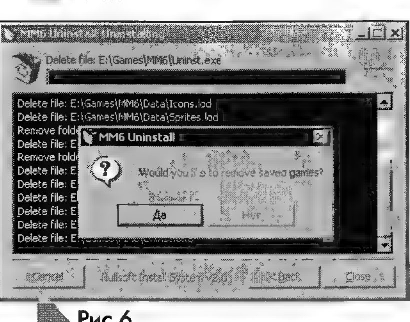


Рис.6

```
##### Uninstall page configuration command:
 UninstallText "$\rYou are about to uninstall
${F_NAME} from your computer"
```

Здесь заканчивается то, что касается инсталлятора, и начинается часть, относящаяся к деинсталлятору.

```
Section "Uninstall"
 SetDetailsView show
 DeleteRegKey HKLM "Software\New World
Computing\Might and Magic® VI\1.0"
 DeleteRegKey HKLM "Software\New World
Computing\Might and Magic® VI"
 Delete "$SMPROGRAMS\3DO\Might and Magic
VI\${S_NAME}.lnk"
 Delete "$SMPROGRAMS\3DO\Might and Magic
VI\Uninstall ${S_NAME}.lnk"
 RMDir "$SMPROGRAMS\3DO\Might and Magic VI"
 RMDir "$SMPROGRAMS\3DO"
 Delete "$DESKTOP\${S_NAME}.lnk"
 Delete "$INSTDIR\Anims\Anims1.vid"
 Delete "$INSTDIR\Anims\Anims2.vid"
 RMDir "$INSTDIR\Anims"
 Delete "$INSTDIR\Data\Bitmaps.lod"
 Delete "$INSTDIR\Data\Games.lod"
 Delete "$INSTDIR\Data\Icons.lod"
 Delete "$INSTDIR\Data\Sprites.lod"
 Delete "$INSTDIR\Data\New.lod"
 RMDir "$INSTDIR\Data"
 Delete "$INSTDIR\Sounds\Audio.snd"
 RMDir "$INSTDIR\Sounds"
 Delete "$INSTDIR\GameUp.exe"
 Delete "$INSTDIR\MM6.exe"
 Delete "$INSTDIR\mss32.dll"
 Delete "$INSTDIR\smackw32.dll"
 Delete "$INSTDIR\readme.txt"
 Delete "$INSTDIR\Uninst.exe"
 MessageBox MB_YESNO|MB_ICONQUESTION "Would you like
to remove saved games?" IDNO +4
 Delete "$INSTDIR\Saves\*.*"
 RMDir "$INSTDIR\Saves"
 RMDir "$INSTDIR"
SectionEnd
```

Здесь, я думаю, интуитивно понятно должно быть почти все. Как и в случае с инсталляцией, включаем режим детального показа проводимых операций, после подтверждения пользователем (**рис. 5**) удаляем сразу целую ветвь из реестра. Далее с помощью серии команд Delete и RMDir удаляем сначала ярлыки, потом все файлы и папки, что мы накопировали (плюс еще некоторые файлы, которые игра сама создает в процессе своей работы). Честно говоря, можно было вместо всех этих команд удаления написать одну — RMDir /r "$INSTDIR", — тогда все, что находилось в инсталляционной папке, включая и ее саму, было бы удалено «за один присест», но... Лично у меня во время игры в этой папке скапливается достаточное количество «левых» файлов — собранный игровой материал, так сказать. Поэтому удаление того, что не записывал инсталлятор, считаю неправильным. В конце деинсталлятор удаляет сам себя (да, этот трюк проходит ☺!), затем спрашивает, нужно ли удалить и записанные игры (**рис. 6**). При отрицательном ответе работа деинсталлятора заканчивается, при положительном — удаляется все, что еще не успел удалить ☺.

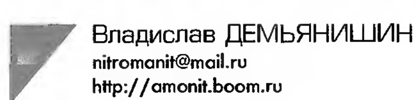
```
; End of installation
```

Что мы получили в результате? Простой, красивый и удобный инсталлятор любимой игры, за работу которого можно ручаться (поскольку все сделано собственными руками), не опасаясь никаких подвохов. Кроме того, поскольку копирование основных файлов не является обязательным компонентом для выбора, наш инсталлятор может к уже проинсталлированной игре добавить недостающий видеофайл для ускорения игры, а также создать ярлыки (а вот такой функциональности я не видел ни в одном инсталляторе ☺). Вот что значит «сделано для себя»!

Вот и все. Я думаю, теперь написать инсталляционный скрипт любой сложности для тех, кто дочитал до этого места, — пара пустяков. Хочется надеяться, что вскоре после выхода в свет журнала со статьей я увижу в Интернете множество удобных, функциональных, компактных и просто красивых программ от читателей МК — с инсталлятором, созданным с помощью NSIS!



Мысли о Паскале



Владислав ДЕМЬЯНИШИН
nitromanit@mail.ru
http://amonit.boom.ru

Продолжение, начало см. в МК, №46, 51–52, 4, 6–7, 10, 12–13, 16–18, 22, 24, 29, 34, 41, 46, 4, 6, 17, 21, 23, 28, 30, 32, 39, 42, 45, 47, 52, 2, 7, 18–19, 23 (165, 170–171, 175, 177–178, 181, 183–184, 187–189, 193, 195, 200, 205, 212, 217, 227, 229, 240, 244, 246, 251, 253, 255, 262, 265, 268, 270, 275, 277, 282, 293–294, 298)

Средства объектно-ориентированного программирования

При каждом удобном случае всякий раз твердят о «трех китах» объектно-ориентированного программирования (ООП) — трех важнейших принципах, коими являются инкапсуляция, наследование и полиморфизм. Рассмотрим-ка поближе этих диковинных зверей.

Инкапсуляция — это объединение данных и подпрограмм обработки этих данных в единую программную структуру (объект). В ООП данные называются *полями (fields)* объекта, а подпрограммы — *объектными методами (methods)*.

Инкапсуляция обеспечивает в широкой степени изоляцию объекта от внешнего окружения и существенно повышает надежность разрабатываемых программ, поскольку локализованные в объекте подпрограммы обмениваются с программой сравнительно небольшими объемами данных, причем количество и тип этих данных обычно тщательно контролируются. Как следствие, замена или модификация методов и данных, инкапсулированных в объект, как правило, не влечет за собой плохо прослеживаемых последствий для программы в целом (из соображений защиты в ООП почти не используются глобальные переменные).

Еще одним немаловажным достоинством инкапсуляции является легкость и простота обмена объектами, переноса их из одной программы в другую. В какой-то мере ООП побуждает программистов к разработке *библиотек объектов*.

Объект — это структура данных, по виду напоминающая комбинированный тип (запись), но может содержать не только поля, но и методы (процедуры и функции).



Описание типа объекта выглядит почти так же, как и описание составного типа, однако вместо служебного слова **record** используется служебное слово **object**:

```
type
  TLine = object
  Left, Top, Right, Bottom : integer;
  Enabled : boolean;
  end;
```

где **Left**, **Top**, **Right**, **Bottom**, **Enabled** — поля объекта. Такой объект может хранить, например, координаты **Left**, **Top** и **Right**, **Bottom** вершины отрезка на экране, и статус видимости **Enabled** на экране. Это пример простейшего объекта, и пока преимуществ над обычными записями не видно. В отличие от записи, объект может иметь *подпрограммы*, управляющие его поведением в соответствии со значениями его полей. Тогда можно объявить метод **Create(Left, Top, Right, Bottom : integer)** для начальной инициализации объекта. При этом имена входных параметров подпрограммы **Create** совпадут с именами полей объекта, что может привести к ошибке компиляции *Error 4: Duplicate identifier (Left)* (продублирован идентификатор Left). Поэтому логичнее будет все имена полей объекта озаглавливать буквой **F** (от слова *Field* — «поле»), тогда объявление объекта будет выглядеть по-другому:

```
type
  TLine = object
  FLeft, FTop, FRight, FBottom : integer;
  FColor : word;
  FEnabled : boolean;
  procedure Create( ALeft, ATop, ARight, ABottom :
integer );
  procedure SetColor( AColor : word );
  procedure Enable;
  procedure Disable;
  procedure Move( ALeft, ATop : integer );
  function Left : integer;
  function Top : integer;
  function Right : integer;
  function Bottom : integer;
  function Enabled : boolean;
  end;
```

В объявлении типа объекта, как и в данном объявлении, заголовки методов указываются после объявления всех полей объекта. И раз уж в объявлении указаны заголовки методов объекта, то далее в листинге следует обязательно указать полное описание методов, при этом имя метода должно состоять из имени типа объекта и собственного имени метода, разделенных точкой (по аналогии с комбинированными типами):

```
procedure TLine.Create( ALeft, ATop, ARight, ABottom
: integer );
begin
FLeft := ALeft;
FTop := ATop;
FRight := ARight;
FBottom := ABottom;
FColor := White;
FEnabled := false;
end;
procedure TLine.SetColor( AColor : word );
begin
FColor := AColor;
end;
procedure TLine.Enable;
begin
if FEnabled then exit;
FEnabled := true;
Graph.SetColor( FColor );
Line( FLeft, FTop, FRight, FBottom );
end;
procedure TLine.Disable;
begin
if not FEnabled then exit;
FEnabled := false;
Graph.SetColor( GetBkColor );
Line( FLeft, FTop, FRight, FBottom );
end;
procedure TLine.Move( ALeft, ATop : integer );
var dx, dy : integer;
begin
Disable;
dx := FRight – FLeft;
dy := FBottom – FTop;
FLeft := ALeft;
FTop := ATop;
FRight := FLeft + dx;
FBottom := FTop + dy;
Enable;
end;
function TLine.Left : integer;
begin
Left := FLeft;
end;
...
function TLine.Enabled : boolean;
begin
Enabled := FEnabled;
end;
```

Причем в каждом методе объекта считаются известными (видимыми) все поля этого объекта. Такое слияние информации об объекте (полей) и методов его поведения (функционирования, деятельности) в едином объектном типе позволяет создавать замкнутую, законченную среду для виртуализации прототипов практически любой сложности, начиная от простой линии на экране дисплея или обычного электронного регулировщика дорожного движения (светофора) вплоть до сложнейших молекулярных и галактических систем.

То, что было, вы наблюдали выше, называется не самим объектом, а лишь *объектным типом*. Чтобы стало возможным применять свойства объекта, необходимо создать *экземпляр объекта*, объявив переменную объектного типа:

```
var
  Line : TLine;
```

после чего можно обратиться (вызвать, выполнить) любой метод объекта, указав составное имя метода, составленное по аналогии с селектором записи. Но необходимо помнить, что слева от точки всегда должна находиться переменная объектного типа, а справа — имя метода или поля:

```
begin
Line.Create(100,100,200,200);
Line.Enable;
Line.Move(200,100);
end.
```

Если таких обращений к элементам объекта много, то, по аналогии с записями, для объектов допустимо применение *оператора присоединения*. Тогда данный фрагмент кода будет выглядеть так:

```
begin
with Line do begin
  Create(100,100,200,200);
  Enable;
  Move(200,100);
  end;
end.
```

Следует заметить: несмотря на то, что к полям объекта можно обращаться так же непосредственно, как и к его методам, это желательно проделывать только внутри его методов, так как обращение к полю объекта напрямую из программы является отступлением от объектно-ориентированного стиля программирования. Вообще, следует осуществлять все манипуляции с полями объекта только через его методы. Следующий код эквивалентен вызову метода **Line.Create(100,100, 200,200)** и не вызовет ошибки, но в дальнейшем лишит программу гибкости, так как реализация метода **Create** может претерпеть изменения, дополнения, а данный код будет работать все так же по старинке.

```
begin
with Line do begin
  FLeft := 100;
  FTop := 100;
  FRight := 200;
  FBottom := 200;
  FColor := White;
  FEnabled := false;
  end;
end.
```

Поэтому логичнее вызывать метод, вместо того, чтобы изменять значения полей объекта непосредственно из программы. К тому же вызов метода выглядит короче, что повысит читабельность программы.

Используя идентификатор объектного типа в **var**-блоке, можно объявить сколь угодно экземпляров объекта.

Из всех подпрограмм, описанных в листинге программы и используемых ею модулях, в результативный код войдут только те, к которым есть обращения (вызовы). Аналогична ситуация и с методами объектов, то есть в результативный код войдут только вызываемые методы объекта, что позволит повысить эффективность программ.

В дальнейшем для тестирования объектов будем использовать следующую программу:

```
Uses profiler, Graph;
var GraphDriver, GraphMode, ErrorCode : Integer;
type
  TLine = object
  ...
procedure InitGrp;
begin
{драйвер и режимы по умолчанию}
GraphDriver := Detect;
{драйвер egavga.bgi в текущей папке}
InitGraph(GraphDriver, GraphMode, 'egavga.bgi');
ErrorCode := GraphResult;
{обработка ошибки}
if ErrorCode <> grOK then begin
 Writeln('Graphics error: ',
GraphErrorMsg(ErrorCode));
 CloseGraph;
 Halt(1);
 end;
end;
var
 Line : TLine;
 j : integer;
Begin
InitGrp;
Line.Create(100,100,200,200);
Line.Enable;
readln;
for j:=1 to 100 do begin
  Line.Move(100+j,100);
  {временная задержка 25 мс}
  NewDelay(25);
  end;
readln;
CloseGraph;
End.
```

где для вывода на графический экран будем использовать возможности модуля **Graph** из пакета *Turbo Pascal*.

При запуске программы на экране появится белая линия, а после нажатия на клавишу *Enter* линия будет плавно двигаться вправо. Выполнение программы завершается повторным нажатием той же клавиши.

Окончание ***на стр. 43***

С машиной на «ты»



Nick 'nVr'
nvr4d@rambler.ru

Ранее в «Моем компьютере» было опубликовано много статей, касающихся программирования с помощью языков высокого и сверхвысокого уровней — тот же Паскаль, Delphi, VB, C++, PHP, ASP... Список можно продолжать и дальше. При этом авторы обходят своим вниманием один из самых важных аспектов программирования — программирование системное. Бесспорно, некоторые фрагменты ассемблерных программ ранее встречались — но систематических материалов по данной тематике на моей памяти еще не было. Что ж... этот цикл статей призван хотя бы в небольшой мере восполнить пробел в области системного программирования.

Часть 1. Ломаем стереотипы

Конечно же, в рамках журнала нельзя полностью рассказать о многогранном — и сложном — искусстве системного программирования. Поэтому я поставил перед собой другую задачу — развеять, или хотя бы попытаться развеять предубеждение подавляющего большинства программистов-«прикладников» против языка Ассемблера, и рассказать об основах методологии построения программ.

Сразу же прошу прощения у тех, кто любит работать с языком Си — у меня с ним хронически не складываются отношения, поэтому иногда трудно бывает воздерживаться от резких (и зачастую необоснованных) выпадов в его сторону.

Прежде чем приступать непосредственно к изложению материала, позволю себе оспорить несколько наиболее типичных предрассудков.

1. «Ассемблер» — святая святых в программировании — очень сложен, и лезть туда совершенно не стоит, когда под рукой есть... (далее — внушительный список интегрированных средств разработки).

Данная точка зрения правильна лишь отчасти. К сожалению, разработчики аппаратных средств, почти ежедневно «балующие» нас технологическими новинками, ей всячески потакают. Ассемблер имеет очень широкий, хотя и ограниченный диапазон применимости. Это то, что мы ранее называли «Системное программное обеспечение». Я имею в виду настоящее СПО: BIOS, загрузчики, утилиты той же ОС ДОС... Также он может применяться и в высококачественных прикладных программах, зачастую в виде вставок. Согласен, иногда программировать сложно — особенно начинающим, но, поверьте моему опыту, игра стоит свеч.

2. Языки высокого уровня и без того неплохо оптимизируют код.

С этим утверждением я столкнулся на... лабораторной работе по системному программированию. Не верьте. Ни одна среда разработки не способна справиться с этой задачей хотя бы вполовину способностей человеческого мозга. Это я проверял, анализируя дизассемблированный код одних и тех же программ на Си, Паскале и Ассемблере. А смотря на размер программ Windows, написанных на C++ и Delphi, впору хвататься за голову. Оптимизация программ — задача нетривиальная, и компьютер не в состоянии пока с ней справиться. Именно поэтому системщики ВСЕГДА будут в цене.

3. Ассемблерщики-профи бывают слегка «не от мира сего».

Факт. Ассемблер заставляет человека, понявшего его, вырабатывать новый стиль мышления. Он, если хотите, облагораживает программиста. Это проявляется обычно в точности, целеустремленности и продуманности действий, а иногда — и в уровне культуры ☺. Что ж... об этом — чуть дальше.

Часть 2. Основы основ

Что же дает нам владение языком Ассемблера? Ответов здесь много.

1. Полный контроль над работой программы, процессора и операционной системы.

2. Возможность оптимизировать программу на самом низком уровне и, как следствие, создавать самые быстродействующие программы.

3. Возможность писать программы, работающие без сторонней ОС.

4. Привычку искать выход из любой ситуации методом «от сложного — к простому», умение разбивать задачу на простые решаемые подзадачи и тому подобные навыки.

Конечно же, чтобы в полной мере воспользоваться этими преимуществами, необходимо разбираться еще и в архитектуре ПК. Но если вы поймете Ассемблер, эти знания придут почти сами собой — справочники с радостью раскроют для вас свои секреты.

Да. Еще мой вам один совет. Привыкайте пользоваться шестнадцатеричной и двоичной системами счисления. Когда придется работать с отдельными битами байта, это будет очень даже кстати.

Вначале мы рассмотрим программную модель микропроцессора (МП) *Intel 80286* в режиме реальных адресов. Почему выбран именно этот процессор и этот режим? Во-первых, потому, что этот МП имеет довольно развитую систему команд, а во-вторых, изучить его программирование куда проще, чем программирование более старших моделей. Естественно, ассемблерные программы, написанные для 286, будут работать и на них. *Режим реальных адресов* избран в силу своей наглядности. В этом режиме работает ОС ДОС и большинство программ под нее, написанных до 1993 года. Он позволяет системе увидеть 1024+64 Кб памяти, выполнять одновременно одну задачу и ни о чем остальном не заботиться ☺. Начнем.

Программная модель МП — это процессор с точки зрения программы, т.е. она описывает возможности, им предоставляемые. Что же может нам предложить «двоечка»?

Во-первых, регистры. Их у нас аж 14.

А. Регистры общего назначения (8 шт). Их можно применять практически для чего угодно. Они включают в себя следующие:

✓ **AX, CX, DX, BX** (*Accumulator, Counter, Data, Base*) — регистры размером в 2 байта. Каждый из них делится еще на 2 однобайтных регистра (например, для **AX** это **AH** (*HIGH* — старший байт) и **AL** (*Low* — младший). Те же правила действуют и для остальных). Еще раз повторюсь, эти регистры подходят для почти ЛЮБЫХ действий, но есть команды, «заточенные» под определенные регистры;

✓ **SP**, **BP** (*Stack pointer, Base pointer*) — двухбайтные регистры. Также могут быть использованы в любых целях, но обычно применяются для работы со стеком (структура типа «последним зашел — первым вышел», работа с ней будет описана чуть дальше). Разница между ними в том, что **SP** реализует «истинно» стековую адресацию, а **BP** позволяет адресоваться к стеку как к обычной памяти;

✓ **SI**, **DI** (*Source Index, Destination Index* — индекс источника, индекс приемника). Существует целый класс команд, называющихся *строковыми*, которые позволяют очень быстро выполнить различные операции с массивами. Главным образом эти регистры предназначены для них, но применяются во многих целях.

Б. Сегментные регистры (4 шт). Используются с одной лишь целью: адресация к оперативной памяти. (*Сегмент* — участок оперативной памяти размером (в нашем случае) 64 Кб. В разделе «адресация» об этом сказано подробнее.) Программно доступны в реальном режиме следующие:

✓ **DS**. Сегмент данных. Здесь хранятся все, если можно так выразиться, ресурсы программы — сообщения, переменные и т.п. Также в нем запроста может быть и исполняемый код, но это далеко не нормальная ситуация;

✓ **ES**. Дополнительный (*Extended*) сегмент данных. Используется в случае, если не хватило 64 Кб для размещения данных, или если необходимо работать сразу с двумя сегментами (например, в тех же строковых командах);

✓ **SS**. Сегмент стека. Процессор имеет возможность организовать часть оперативной памяти в виде стека, и «правильная» адресация к стеку возможна лишь через **SS**.

Особняком стоит регистр **CS**, отвечающий за сегмент кода. «В лоб» его значение выставить нельзя, да и бессмысленно (его непродуманное изменение приводит, как правило, к зависанию машины из-за попадания на конвейер МП «левых» команд). Однако повлиять на него можно — например, с помощью стека.

В принципе, ничто не запрещает, чтобы несколько сегментных регистров указывали на один и тот же сегмент. Более того, часто так и делают.

Есть еще 2 спецрегистра, которые напрямую недоступны даже для чтения. Это регистр **IP** (*Instruction pointer* — регистр, указывающий, какой байт памяти должен загрузить процессор на выполнение), и **FLAGS** — текущие флаги МП. На эти регистры тоже можно влиять через стек, командами перехода, на **FLAGS** же влияет чуть не каждая вторая команда. Все регистры также двухбайтные; флаги (биты регистра **FLAGS**) интерпретируются следующим образом:

✓ **0 CF**, *Carry Flag*. Устанавливается в **1**, если произошел перенос из старшего разряда;

✓ **1** не используется, установлен в **1**;

✓ **2 PF** — *Parity flag*, флаг четности. Устанавливается в **1**, если младший байт результата содержит четное число единиц;

✓ **3** не используется, установлен в **0**;

✓ **4 AF** — *auxiliary carry*, применяется для операций с двоично-десятичными числами;

✓ **5** не используется, установлен в **0**;

✓ **6 ZF** — *zero flag*, устанавливается в **1**, если операция имеет нулевой результат;

✓ **7 SF** — *sign flag*, флаг знака. Отображает состояние старшего бита результата;

✓ **8 TF** — *trace flag*, флаг трассировки. Процессор переходит в режим покомандного выполнения (прерывается после каждой команды). Так работают многие отладчики;

✓ **9 IF** — *interruption flag*, флаг прерываний. Если установлен в **1**, прерывание работы процессора аппаратурой разрешено;

✓ **A DF** — *direction flag*, флаг направления. Используется в строковых командах;

✓ **B OF** — *overflow flag*, флаг переполнения. МП устанавливает его в **1**, если результат не помещается в месте назначения;

✓ **C, D, E** — не используются в режиме реальных адресов;

✓ **F** — не используется, установлен в **0**.

Для флагов **CF**, **IF**, **DF** существуют специальные команды, устанавливающие и сбрасывающие их. Мнемоника команд такова:

✓ установка в **1** — **ST[первая буква имени флага]** (**STC** — *SeT CF*);

✓ установка в **0** — **CL[аналогично]** (**CLC** — *CLear CF*). Остальные флаги необходимо устанавливать вручную.

С регистрами вроде бы как разобрались. Если сейчас что-то неясно с теми же флагами, ничего страшного — по мере практического использования все прояснится. Теперь — память.

В режиме реальных адресов используется *сегментная адресация памяти*.

Сегмент, как уже говорилось — участок памяти размером 64 Кб. Исходя из размера сегментных регистров, можно сказать, что в памяти существует 65 536 сегментов. Как же они размещены в 1 Мб памяти? Очень просто — они перекрывают друг друга каждые 16 байт. Физический 20-разрядный (посчитайте, сколько памяти так адресуется) адрес памяти формируется так: берется номер сегмента, умножается на 16, и к нему прибавляется смещение. Например, нам надо обратиться к 32 767 байтам памяти. Адрес в форме **сегмент:смещение** можно записать так:

```
0000:32767 (0*16+32767 = 32767)
```

А можно и так:

```
0001:32751 (1*16+32751 = 32767)
```

А в нормальной форме — **2047:0015**

Вообще, процессоры серии x86 по праву считаются одними из самых сложных как по реализации, так и по программированию. Для явной адресации к оперативной памяти с помощью регистров тут отведено целых 24 способа. Указателями могут быть регистры **BX**, **BP**, **SP**, **SI** и **DI**. Например:

```
MOV AX, BX ;помещаем в AX значение BX
MOV AX, ES:[BX] ;помещаем в AX 2 байта по адресу ES:BX. Если ES=10, BX=2000, то этот адрес равен 10*16+2000=2160.
```

Все регистры-указатели имеют сегментный регистр, относительно которого они адресуются «по умолчанию». Для **BX** и **SI** это регистр **DX**, для **BP** — **SS**, для **DI** — **ES**. То есть команды **MOV AX, DS:[BX]** и **MOV AX, [BX]** будут эквивалентны. Использование второго типа команд предпочтительно, так как команда занимает меньше места и выполняется быстрее.

Адресоваться явным образом к оперативной памяти компьютера также можно с помощью всевозможных комбинаций вышеперечисленных регистров и целых чисел. Допускается использование не более двух регистров одновременно. Но выйти за пределы сегмента не получится — устройство, вычисляющее смещение, имеет разрядность в 16 бит.

Примеры:

```
[BP+SI]
[BX+SI+140]
[BX+4000]
```

и т.п.

Существует также куча способов адресоваться к памяти неявно, без указания регистров. Но о них я расскажу в следующий раз.

С памятью вроде бы тоже более-менее понятно, по крайней мере для начала. Конечно, есть еще интересный способ для 286 процессора получить дополнительные 64 Кб памяти выше 1 Мб, но поскольку это скорее тема отдельной статьи, рассматривать его мы пока не будем.

Еще очень интересная вещь, особенно для тех, кто хочет работать с аппаратурой, — *адресное пространство портов ввода-вывода*. Это — часть адресного пространства ПК, размером в 64 Кб, к которому, грубо говоря, подключается своими интерфейсами оборудование. Каждый порт эквивалентен 1 байту, т.е. содержит 8 бит. В Windows, когда вы просматриваете сведения об устройствах, во вкладке *Ресурсы* зачастую присутствует параметр *диапазон ввода-вывода*. Это и есть совокупность портов, используемых устройством. С вводом-выводом есть одна проблема: по сравнению с ЦП периферийные устройства, как правило, медленны. Поэтому при многократном вводе-выводе следует вставлять дополнительные команды ожидания *NOP*. Пары команд, как правило, хватает даже на 2-ГГц Pentium 4.

Наиболее характерные диапазоны — *3F8h*, *2F8h* (COM1, COM2); *378h* (LPT1), *220h* (Sound Blaster), *42h–45h* (системный таймер) и пр. Подробные описания портов приведены в многочисленных книгах по архитектуре ПК. Это — один из самых низких уровней программирования. Большинство устройств имеют свою собственную систему команд и программируются непосредственно через соответствующие IO-порты — как, например, контроллеры гибких/жестких дисков, видеоадаптеры. Т.е. программист работает с машинным кодом специализированных процессоров. По счастью, системы команд у них намного проще, чем у ЦП.

Таким образом, знания об IO-портах пригодятся вам, если вы соберетесь писать собственные драйверы устройств. К сожалению, несмотря на массу книг по соответствующей тематике, многие вещи документированы весьма плохо, и поиск приходится производить самостоятельно.

В следующей статье я планирую рассказать об основных командах и директивах Ассемблера с тем, чтобы вы уже могли писать первые свои программы.

Если у вас возникли какие-то вопросы — пишите, я с радостью отвечу.

Да, чуть не забыл самое главное... Литература. Конечно же, это *Assembler: учебник для вузов* и *Assembler: специальный справочник* **Виктора Юрова**. С этих книг я, собственно, начал — не учить, нет, ПОНИМАТЬ Ассемблер. И именно благодаря им он стал одним из моих любимых инструментов. Для тех же, кто захочет в большем объеме изучить архитектуру IBM PC, рекомендую книгу нашего преподавателя **А. И. Поворознюка** *Персональные компьютеры IBM: Архитектура и управление*, которую можно взять с **http://khpi.narod.ru**, или классику жанра — *Справочник программиста ПК типа IBM PC, XT и AT* **Роберта Джордейна**. В последнее время появилось, конечно, много интересных книг по этому вопросу, но они, во-первых, стоят дорого, а во-вторых, довольно сложны в изучении.

(Продолжение следует)

Больные и их болезни

Краткий медицинский справочник

Ольга КАЛИТКА
ok_best@inbox.ru

Окончание, начало см. в МК, № 38 (313)

Чайники *(происхождение слова медицине неизвестно)*

Симптомы. Малознающие пользователи. Часто не знают, где находится кнопка включения компьютера, хранят нужные документы в «Корзине» и используют компакт-диски только как удобное зеркальце. Легко закипают, но так же быстро остывают.

Лечение. Болезнь легко поддается лечению при помощи разъяснительных бесед с гуру и чтения специальной литературы. Сейчас выпущена целая серия лечебных книг, в названии которых указано: «Для чайников».

Осложнения и прогноз. Больной либо полностью выздоравливает и больше никогда не подходит к компьютеру, либо становится юзером, что благоприятно сказывается на его тонусе и долголетии. При осложнениях возможно превращение в ламера.

Геймеры *(от англ. «game» — «игра»)*

Симптомы. Часто страдают маниями преследования или повышенной агрессивностью. Могут говорить о компьютерных играх часами, переходя на малопонятный сленг. Геймерам сложно адаптироваться к реальной жизни, они верят в запасные жизни и сохранение. Это дает повод говорить о болезни, осложненной религиозными представлениями геймеров.

Лечение. Лечение длительное и дорогостоящее. Лучший способ — отформатировать винчестер компьютера, за которым часами просиживает геймер. Как вариант — сжечь монитор.

Осложнения и прогноз. Даже после выздоровления нужно контролировать поведение бывшего геймера, так как часто болезнь возвращается. Прогноз неблагоприятный.

Сисадмины *(от англ. «system administrator» — «системный администратор»)*

Симптомы. Управляют компьютером и юзерами. Часто издеваются над своими подопечными, закрывая им доступ к играм и порносайтам. Не ладят с чатерами, форумчанами и серферами. Иногда страдают манией величия.

Лечение. Лечение неэффективно, так как всегда найдется здоровый человек, который заражается вместо выздоровевшего. Врачам во избежание заражения рекомендуется носить ватно-марлевую повязку и пользоваться свежим антивирусом.

Осложнения и прогноз. Прогноз крайне неблагоприятный. Болезнь обычно протекает без осложнений и становится хронической. Иногда сисадмин вылечивается, но тут же становится гуру.

Юзеры *(от англ. «user» — «пользователь»)*

Симптомы. Управляют компьютером, а компьютер управляет ими. Умеют пользоваться основными программами, и в случае зависания ловко перезагружают компьютер нажатием кнопки «Reset». Часто отмечается паническая боязнь сисадминов.

Лечение. Болезнь не требует лечения, но нужно внимательно следить, чтобы она не перетекала в иные формы. Для юзеров медицина делает больше, чем для инвалидов: созданы специальные книги и программы с понятным интерфейсом.

Осложнения и прогноз. Прогноз в целом благоприятный.

Программисты (от англ. «program» — «программа»)

Симптомы. Больные одержимы желанием что-то сотворить. Вид не менее творческий: красные глаза, небритое лицо и трясущиеся руки. Изъясняются на непонятных языках. Могут не вставать из-за компьютера до трех суток. В группу риска входят мужчины от 16 до 40 лет. Употребление пива и курение повышает риск заболевания.

Лечение. Исключить все вредные привычки, а именно: крепкий кофе рядом с клавиатурой, курение у монитора, потребление пива в дозе более 0.33 л. Для лучшего эффекта удалить все созданные больным программы с компьютера вместе с отладчиками кода и компиляторами. В запущенных случаях программист продолжает писать код на бумаге — тогда можно говорить о неизлечимости больного.

Осложнения и прогноз. При отсутствии лечения программист может стать хакером.

Уеб-дизигнеры *(от англ. «web» — «паутина», «designer» — «конструктор»)*

Симптомы. Как и программисты, страдают от постоянного желания что-то сотворить, или, как они это называют, «задизайнерить». Проявляют явные эксгибиционистские наклонности, так как выставляют сотворенное напоказ на всеобщее обозрение в Интернете. Своим родным языком считают HTML. Страдают отсутствием художественного вкуса, часто развивается дальтонизм.

Лечение. Больному нужно прививать чувство меры и рассказывать о несовместимости красного фона и зеленого шрифта. Неплохие результаты дает также общение с гуру.

Осложнения и прогноз. При правильном лечении прогноз благоприятный — уеб-дизигнер излечивается и становится здоровым веб-дизайнером. Повторное заражение исключено. При отсутствии лечения больной рискует стать жертвой возмущенных серферов.

Кулхацкеры *(от англ. «cool» — «крутой», «hack» — «рубить»)*

Симптомы. Любят ломать простенькие программы. Мечтают найти крякер Интернета или как минимум хакнуть какого-нибудь провайдера. Отыгрываются на ламерах и чайниках, иногда умудряются заразить юзеров. Больные маскируются под хакеров, вводя в заблуждение даже врачей. Чаще всего заболевают молодые люди в возрасте от 12 до 1В лет.

Лечение. Обычно заболевание проходит с возрастом.

Осложнения и прогноз. Осложнения пока не наблюдались.

Хакеры *(от англ. «hack» — «рубить»)*

Симптомы. Симптоматика еще до конца не изучена. Впрочем, современная медицина может утверждать: хакеров либо не знает никто, либо знает весь мир. Они взламывают крупные сайты, пишут кряки для платных программ и делают бесплатной даже операционную систему Windows.

Лечение. Обычно лечением занимаются не врачи, а уполномоченные органы. Эффективность проводимой ими терапии достигает 100%.

Осложнения и прогноз. Прогноз в каждом случае индивидуальный. Срок принудительного лечения определяется Конституцией и судом.

Спамеры (от англ. «spam» — название мясной консервы)

Симптомы. Рассылают рекламные письма всем знакомым и незнакомым интернетчикам, за что и те и другие их сильно не любят. Любят маскировать свои реальные адреса. В оффлайновой жизни никогда не сознаются в своей болезни.

Лечение. Самое сложное — привести больного на прием к врачу, так как обычно спамеры избегают посещать поликлинику и вообще появляться на людях. Когда диагноз подтвержден, назначается суровое лечение: отключение от Интернета. Некоторые врачи также пытаются лечить спамеров штрафами.

Осложнения и прогноз. Прогноз при своевременном лечении благоприятный. Могут наступить серьезные осложнения при встрече с серферами, также небезопасно для здоровья общение с гуру, юзерами и программистами.

Гуру *(происхождение слова уходит в религиозные дебри)*

Симптомы. Весь внешний вид и поучающая манера разговора гуру выдает в них всезнающих и умудренных жизнью людей. Больные готовы делиться своим богатым опытом, забывая о еде и сне. Страдают манией величия, которую стимулируют другие больные. Люди нередко путают их с хакерами, и только опытный врач может точно поставить диагноз.

Лечение. Носит профилактический характер. Гуру нужно по возможности изолировать от чайников и ламеров, несмотря на стремление больного общаться с ними. Также полезен свежий воздух, общение с форумчанами и другими гуру.

Осложнения и прогноз. При своевременной профилактике прогноз благоприятный.

Серферы *(от англ. «surf» — «прибой»)*

Симптомы. Больные отличаются патологической любознательностью, а также непостоянством, так как они постоянно бросают один сайт и уходят к другому. Болезнь активизируется ночью, когда цены на повременный доступ в Интернет падают.

Лечение. Лечится вводом оплаты за трафик. Если болезнь продолжает развиваться, применяют ампутацию модема. В тяжелых случаях врач может принять решение об удалении телефонной линии.

Осложнения и прогноз. При строгом лечении прогноз благоприятный. При склонности к осложнениям больной может стать чатером или форумчанином, поэтому необходим постоянный врачебный контроль.

*Окончание. **Начало на стр. 38–39***

Наследование и переопределение

Итак, ранее мы сконструировали объект **TLine**, управляющий линией на экране дисплея — и это получилось у нас весьма недурственно. Но что делать, если нам понадобится создать объект для управления прямоугольником — такой себе рамкой на экране? Что же нам, опять составлять новый объектный тип и опять переписывать кучу кода? Нет уж, давайте воспользуемся *наследованием*. Что это такое? О, это такая фишка — всем фишкам фишка ☺. Ну а если серьезно, то в Turbo Pascal имеется возможность объявлять новые объектные типы в качестве *потомков* уже существующего объектного типа, что позволяет потомкам унаследовать все поля и методы родителя (предка), а также дополнять объекты новыми полями, заменять (переопределять, перекрывать) методы родителя или дополнять их. Вот это, собственно, и есть наследование.

Чтобы описать объект-потомок, достаточно после служебного слова **object** указать в круглых скобках *идентификатор типа* объекта-родителя:

```
type
  TFrame = object (TLine)
   FFilled : boolean;
  function Filled : boolean;
  end;
```

это будет означать, что все поля и методы родителя **TLine** присутствуют неявно, и стало быть, присовокупляются к потомку — родная кровь, все-таки ☺. При этом в новый объект можно ввести дополнительные поля и методы, если в этом есть необходимость. Так, в потомка **TFrame** было введено поле **FFilled**, управляющее заливкой рамки, что позволит получать на экране либо незакрашенный прямоугольник-рамку, либо закрашенный прямоугольник а-ля «Черный квадрат» Казимира Севериновича ☺.

В свою очередь, от типа **TFrame** можно образовать («породить») потомка — например, **TWindow**, ведь свойства рамки и окна очень схожи. Тогда объект **TFrame** будет предком по отношению к **TWindow**, а **TWindow** будет прямым потомком объекта **TFrame** и косвенным потомком объекта **TLine**, стало быть, унаследует все поля и методы, которые были описаны в его предках. Такое объектное генеалогическое древо своими ветвями может подниматься высоко и тянуться к небу, пока это будет нужно селекционеру, то бишь программисту. И значит, не имеет смысла снова и снова изобретать велосипед, а остается лишь совершенствовать его и нагромождать необходимые надстройки, дополнения — а это экономит время при разработке программ.

При этом действует правило, что один объектный тип может служить предком для неограниченного числа объектных типов-потомков, но каждый объектный тип может быть потомком только одного родителя, который указывается в скобках.

Вернемся к нашим баранам. Мы остановились на объявлении нового объектного типа **TFrame**, который, помимо всего прочего, унаследовал от своего предка **TLine** методы **Enable** и **Disable**. Да, но ведь эти методы рисуют линию, а объект **TFrame**, по идее, должен рисовать прямоугольник. В этом случае можно поступить по старинке, то есть добавить два метода, которые бы рисовали рамку и назывались другими именами. Но это было бы отступлением от объектно-ориентированного стиля программирования. Более элегантным решением может послужить переопределение методов **Enable** и **Disable**, то есть замена их новыми методами с аналогичными именами; заодно переопределим и метод **Create**, чтобы выполнялась инициализация поля **FFilled**:

```
type
  TFrame = object (TLine)
   FFilled : boolean;
  procedure Create( ALeft, ATop, ARight, ABottom :
integer );
  procedure SetFilled( AFilled : boolean );
  procedure Enable;
  procedure Disable;
  function Filled : boolean;
  end;
procedure TFrame.Create( ALeft, ATop, ARight, ABot-
tom : integer );
begin
{вызываем родительский метод}
TLine.Create( ALeft, ATop, ARight, ABottom );
FFilled := false;
end;
procedure TFrame.SetFilled( AFilled : boolean );
...
```

Пока приведу лишь исходный код метода **Create**, а остальные методы будут рассмотрены позднее.

Выше мы переопределили методы, тем самым перекрыв родительские методы новыми. При этом в объекте-потомке под теми же именами будут известны только его собственные методы, а *унаследованные (inherited)* методы будут доступны посредством указания составного имени, состоящего из имени родительского типа и имени метода, разделенного точкой.

Из нового метода **Create** видно, что в нем используется вызов унаследованного метода **TLine.Create** — чтобы не делать двойную работу. Начиная с версии *Turbo Pascal 7.0* введено служебное слово **inherited**, которое позволяет осуществить вызов перекрытого унаследованного методо. Тогда для седьмой версии метод **Create** может иметь следующий вид:

```
procedure TFrame.Create( ALeft, ATop, ARight, ABot-
tom : integer );
begin
{вызываем родительский метод}
inherited Create( ALeft, ATop, ARight, ABottom );
FFilled := false;
end;
```

Выглядит гораздо лучше.

Помимо этого, у нового метода может быть совсем другой интерфейс входных параметров, чем у одноименного унаследованного метода.

В отличие от методов, поля переопределить нельзя, так как они в обязательном порядке наследуются объектом-потомком. Можно добавлять новые поля, но при этом их имена не должны совпадать с именами полей родителя.

Совместимость объектных типов

Кроме эквивалентных объектных типов, объявленных как **type TObj1=TObj2**, совместимыми по присваиванию являются объектные типы, состоящие в родстве, при этом можно присваивать потомка родителю:

```
var
  Line : TLine;
  Frame : TFrame;
begin
Line := Frame;
...
```

Такое же правило распространяется и на указатели объектных типов. При подобном присвоении значениями полей объекта-потомка модифицируются только одноименные поля объекта-родителя.

(Продолжение следует)

Колонка редакторов

Беседка «Моего компьютера»

Трурль
reader@mycomp.com.ua

Беседка — это место, в котором встречаются читатели всех поколений. Просто тех, кто постарше, мы не так часто видим: народ они, сами понимаете, уже сильно занятый житейскими заботами. Но когда они находят время нам написать, то письма эти интересны всегда. Если они дают советы, те оказываются важными и актуальными для всех. Если рассказывают о жизни, то умеют в краткий формат письма уместить небольшую повесть. Это называется Опыт Прохождения Жизни.

«Привет, Трурль! Мне 35 лет, и в комплекте с радикулитом я имею 2 детей (а, может, они меня?), причем Коля старше Вовы на 5 минут. Теперь, я думаю, вам понятны проблемы домашней компьютеризации.

Так вот, у меня дома стоят 2 персоналки на 486 процессорах. Я до сих пор успешно пользую DOS 6.22 с засаженным на него в качестве графической оболочки WIN 3.11 в комплекте с MS OFFICE 4.3 (откуда на самом деле взят незабвенный MS Word 6.0). В комплекте установлены и успешно работают в течение 2-3 лет IE 3.01 (спецрелиз под 16-разрядную WIN 3.11) со своей почтовой системой, Total Comander, ADSC и куча DOS-программ, включая достаточно красивые игры, типа HMM или WARCRAFT.

В комплекте программного обеспечения 90% русифицировано. Даже моя супруга, человек далекий от этих вопросов, пишет свои рабочие бумажки дома, в кресле, под сигаретку с кофе, а потом по почте на работу шлет.

Машины связаны друг с другом по нуль-модемному кабелю, поэтому у нас в порядке вещей семейные споры решать дуэлью в DOOM или DUKE NUKEM, хотя я предпочитаю HMM на пару с соседом. На более мощной машине (на одной камень 100 мгц, на другой — 80) стоит MP-3 проигрыватель под DOS, после подключения ее к отдельному усилителю получился неплохой магнитофон. Короче, кому интересно, мой адрес прилагается.

И теперь самое забавное: с нами живет бабушка, ну в смысле теща, так вот, когда процесс воспитания любимых внуков заходит в тупик, она просто вырубает сетевые автоматы на счетчике и... ничего не происходит. В DOS-е нет понятия жесткого рестарта системы, там можно вырубать свет в любом месте, кроме сохранялки последней миссии в какой-нибудь игре ничего не теряется. По сравнению с Виндой эта система надежна, как автомат Калашникова...» **Олег Слободнюк** (alinkij11@ukr.net)

Видите, что получается? Обладание домашним компьютером — это больше категория качественная (есть он или нет), чем количественная («У меня пень-четыре...» «Фу, а у меня уже даже шестой!»). Главное — уметь достойно загрузить имеющуюся технику. «На руках», в смысле, в домах у народа еще много описанных устройств, и жаловаться, что они чего-то не могут, и себя при этом старательно жалеть — это, как видите, опрометчиво. Потому что даже вам завидуют все те, у кого компьютера нет вообще.

Уныло дождь стучит на крыше,
Играть в «Макс Пэйн» желанья нет...
Так, может, сесть за мат. анализ?
WarlocK_[RF]

А вообще, дело пользования ретротехники — это область обширная. В ней много зависит от сообразительности, от нормальных рук и от полезной информации, которая попалась на глаза в нужный момент. Так, может, кто из антикваров захочет сделать сайтик МК-манов — владельцев двоек, троек и четверок? А если мэмэхатники к ним попросятся, то и их пустите. Там сможете спокойно обмениваться опытом и выручать друг друга советами.

Если пойдет такая тема, напишите — прорекламируем.

Ахтунг, минен!

К сожалению, рубрика эта периодически появляется на наших тесных страницах. Переживаем мы, что разнообразные информационные напасти подстерегают вас в засадах. Они таятся то за новой соблазнительной железкой, то за пиктограммкой популярной программы. А пока вы с ними боретесь, у вас нет времени писать письма в Беседку. А если даже и пишете, то получается очень грустно.

«Значит, использовал в одном известном пакете утилит (не будем называть его в целях рекламы) программу RegCleaner. Ну, думал, почищу реестр, и все пойдет как по маслу... «Ага, щас»! Перестали фильмы идти... Я уже эти кодеки ставил-ставил... Не помогло. Вот, теперь собираюсь переустанавливаться, если ничего в голову не придет, или кто-то в нее не положит.

Так-то, люди. С реестром шутки плохи! Доказано на практике.

P.S. При проведении опыта пострадал 1 компьютер». **Marcus** (marcus@gala.net)

Читатели, может кто-то уже пострадал от этой напасти и выжил, так поделитесь проходилкой с Маркусом.

Книга рекордов Трурля

«Привет, Трурль!!! Подумал, и решил написать тебе в книгу рекордов, авось пригодится, на тему, сколько я прошел игр. За 2 года существования моего компьютера — 120 штук. Может, для кого это и не рекорд, но я горжусь собой. А если еще и посчитать, сколько я потратил на это сил и денег, но это уже не та история...» **Сергей Булавик**

— А как проходил? На каких уровнях сложности?

— В большинстве случаев на «среднем». А если игра нравилась, то по второму разу на «тяжелом» уровне. На легком, если честно, еще ни разу не пробовал.

Итак, кто больше настрелял, настроил, налетал и наездил?

HKEY_CLASSES_ANIMAL

«Привет, Трурль!!! Была в «Беседке» тема о домашней живности, так вот это — кошка программиста.

Ну, что я могу сказать... Пытается спать не только на мониторе, но и на клавиатуре — когда я что-то пишу. Но под сладкое мурлыканье так хорошо код пишется...» **Tigress**

Напечатать это письмо у нас было несколько поводов.

Во-первых, вспомнился анекдот, который в Safe Mode исполнении звучит так: «Некоторые программисты при написании софта какое-то время прыгают одним местом по клавиатуре. Потом компилируют, что получилось, а затем продают нам с вами за большие деньги».

Интересно, а что может «написать» кошка, если ей дать просто поваляться на клавишах? А если перед этим угостить валерьянкой?

Компьютерные анимальщики, проведите опыты. Вдруг мы все в двух шагах от нового программного прорыва?! Вдруг кошки по коэффициенту полезного действия намного опередят приматов? Помните байку из вузовского курса теории вероятности, что сто обезьян, тупо, но терпеливо нажимая на клавиши, рано или поздно напечатают «Войну и мир»? (Забавно, что в этом опыте даже гипотетически не предполагалось использовать людей...)

Так вот, в конечном варианте желательно было бы получить формулу: 100 обезьян = X кошек = Y программистов = Z Трурлей. Последнее равенство введено специально, чтобы программеры не обижались, что «их тоже посчитали».

Во-вторых, можно прогнозировать, что у наших читателей, при наличии под рукой цифровой и аналоговой фототехники, накопилось уже много любопытных снимков, как на темы домашнего животноводства, так и на иные, связанные с компьютером. Так мы предлагаем вам присылать их в Беседку. Пусть все читатели посмотрят, оценят, выберут лучшее. Победителю конкурса мы в конце года что-то положим под елочку. Так что есть время и покопаться в архивах, и успеть что-то подсмотреть новое и оцифровать.

Ой, смотрите, что приближается!!!

Кстати, о конкурсах. До сих пор в Беседочной почте встречаются письма, в которых вы вспоминаете о прекрасных, волнующих временах, когда мы их с вами проводили. Темы конкурсов отражали самые различные стороны компьютерного бытия. Многие вопросы были связаны с информацией из статей, напечатанных в МК, так что при поисках ответов преимущество имели обладатели домашних журнальных архивов, а также те, кто знает сетевую тропку на наш сайт.

А призов мы по ходу дела пораздавали!!! Главный победитель, между прочим, получил **компьютер**... Нормально, да?!

Так вот, мы решили повторить это приключение.

Готовимся. В ближайших номерах ждите первый конкурс нового цикла. Читайте нас регулярно и покупайте. Учитывайте, что каждый приобретенный в киоске журнал снижает вероятность попадания его к эрудированному конкуренту!

Страна советов

Читаешь ваши письма-советы и убеждаешься, что действительно «...компьютер людям нужен для того, чтобы бороться с проблемами, которые не существовали до его появления...»

А еще «...любое устройство, требующее настройки, с максимальными трудностями поддается этому...» — это уже цитата из другого источника.

Ну почему разработчики программ совершенно не думают об изделиях своих коллег по ремеслу, которые должны одновременно умещаться на компьютере? Может из величия личных планов, не позволяющих присматриваться к мелочам окружающего бытия? Или из вредности?

✓ **Совет №40.** *«Вот, поставили мы антивирусник (AVP, к примеру). Естественно, выборочно, сканер и монитор. И сразу заметно ощущаем тормоза. Но зачем нам монитор, когда мы играемся, набираем реферат или читаем е-буку? Отключаем его из автозагрузки (из реестра или меню «Пуск»). В 98-м этого достаточно. Но в 2000/XP тормоза практически не исчезают. Кликаем правой кнопкой мыши по «Мой компьютер» и выбираем «Управление» («Manage», вроде). В открывшемся окошке по дереву спускаемся в самый низ (кто видит это окошко впервые, советую посмотреть другие пункты — много интересного найдете) и находим «Службы». Вот он куда спрятался, наш AVP! Как видно, служба может быть отключена, включена и находится на ручном управлении. Правый клик по монитору антивирусника — переводим его на ручной запуск. Все!*

Так же я поотключал себе службу принтера, техподдержки и еще несколько ненужных. Но! Думайте, прежде чем что-то менять! Теперь для удобства кидаем ярлычок монитора в «QuickLaunch» и запускаем монитор только когда отправляемся в сеть, что-то устанавливаем, читаем диски и т.п.» **Boba**

Давайте дружить городами

«Приветик, ТРУРЛЬ! У меня вопрос лично к тебе. Слушай, ты не знаешь, есть ли Клуб Любителей МК или что-то подобное в славном городе Кривом Роге? Я читаю журнал 2 года, и в «Беседке» то и дело пишут: «Давайте создадим клуб МК в таком-то городе». Но никогда я не видел, чтобы кто-то про Кривбасс писал. Город большой, и я думаю, что кто-то уже его создал». С уважением, **Radeon** (radeon@alba.dp.ua)

Если его кто-то создал, то читатель, думаю, получит письмо с приглашением на ближайшую party. А если клуба еще нет, Радеон будет рад сам его основать и стать, соответственно, Президентом, ну, или хотя бы Генеральным Секретарем.

«Щас спою!..»

Дело было летом. Однажды (ближе к вечеру) потянуло нас на песни. И затянули мы истинно программистскую песню «Сто шагов назад» популярной группы Мевиагладзе. Подробности происшествия можете глянуть в МК, №21 за 2004 (http://www.mycomp.com.ua/text/6829). Как показало литературоведческое вскрытие текста — произведение это очень точно и образно описывает тяжкие будни тружеников GOTO и старателей IF THEN.

Ну, а потом народ задумался, что это попсовики за нас песни сочиняют?! А сами-то? Компьютерщики же изнутри намного подробнее представляют, что такое труд создателя софта. Ну, вот они, подумав-подумав, и пишут Трурлю: «А сами мы тоже сможем»!

«Давайте», — отвечаю. Чем мы хуже?!

Но, как водится у программеров, чтобы куски кода не пропадали, им всегда находится применение. Ну, вот и новая песня была предложена на старый мотив. Автор Gluck пишет:

«Салют, Трурлик! Прочла в 21-м номере МК отзыв на поп-песню «100 шагов» — правда, смешно! Про нас... А вот если б у песен тоже были версии, как у прог, то слова могли бы быть такие (поэт из меня, правда, не супер, зато типа в музыку попадаю):

100 шагов вперед
Не торопитесь с инсталляцией,
Программа просит регистрации,
Но плод усилий чьей-то логики
Мы будем юзать по-другому

Не будем время тратить попусту,
В И-нете кряк скачаем попросту.
Но самому же интереснее...
К тому ж опять провайдер занят (пи-пи-пи)

100 шагов вперед —
Прога в Soft-Ice'е.
Вот блин, глючный код
Давай, поддавайся!
100 шагов вперед —
И защиты больше нет!

100 шагов вперед
Любимый дебаггер,
Вот тот переход —
Давай его на фиг!
100 шагов вперед —
И защиты больше нет!

Декомпилятор установим-ка,
И от EXE-шника к исходникам...
И плод работы чьей-то логики
Работать будет по-другому!!!

Эээ, что-то песня получилась какая-то хакерская. Не петь ее? Подумали мы и решили, лучше уж подсознательные ломательские желания, живущие в каждом из нас, реализовать в поэзии, чем действительно — брать что-то и вскрывать.

Гм, тем более, что найти сегодня хоть что-то не сломанное так сложно!!! Вокруг столько прозаиков!